MODELING TRUSTWORTHY BEHAVIOR AND LIMITING THE IMPACT OF SELFISHNESS

BY

TIMOTHY STEVEN MURRAY

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Industrial Engineering
in the Graduate College of the
University of Illinois Urbana-Champaign, 2021

Urbana, Illinois

Doctoral Committee:

Professor Rakesh Nagi, Chair and Director of Research
Assistant Professor Jugal Garg, Director of Research
Professor Carolyn L. Beck
Assistant Professor S. Rasoul Etesami
Professor Rayadurgam Srikant

ABSTRACT

The major theme of the research in this dissertation is the modeling of selfish behavior and the mitigation of its effects. Game theory literature asserts that all agents behave with complete self-interest. However, this is at odds with empirical studies in behavioral economics which routinely show subjects engaging in behaviors which allow them to be taken advantage of by other agents. Despite this, the other agents rarely do so. In order to predict when and to what degree agents engage in self-serving actions, we introduce the concept of a Limited-Trust Equilibrium (LTE), a state in which all agents contribute to each other's utility, provided it is not too expensive for them personally. Each agent is motivated to do so in order to inspire reciprocity from its fellow agents and thus benefit in the long term. The LTE is then shown to exist in all finite games, and the utility of agents who play in a limited-trust manner is compared theoretically and numerically to those who play in a purely self-serving manner to illustrate why the agents prefer to interact in this way.

The concept of limited-trust is then applied to a social setting, in which players need to attract and form partnerships in a social network. This induces a metagame in which players must decide how much they are willing to commit to reciprocity in order to attract partners, where players who behave in a less selfish manner are naturally more attractive partners, but more selfish players benefit more per partnership formed. When other factors are not kept equal, such as when not all players are able to provide the same opportunities to their potential partners, we see the emergence of "diva" behavior, in which talented or well-connected players are easily able to form partnerships despite behaving in a mostly or entirely selfish manner. A paper based on this work is nearing its conclusion, and is expected to be submitted prior to Final Defense.

As initially mentioned, our research also touches on the mitigation of the

effects of selfish behavior. A major focus of research in Game Theory is on designing games in which the interests of the players align with the interest of the game's administrator or coordinator, generally maximizing the net utility or minimizing the net cost of the system the game operates in. Therefore, following our work on the LTE to better model when and how selfish behavior occurs, we pivot to focus on this area. We introduce the Prize Collecting Multiagent Orienteering Problem (PCMOP), a Game Theoretic version of the Orienteering Problem with applications to ride-sharing. We show it to be part of the class of valid utility games, then propose and analyze three policies for mitigating selfish behavior in the PCMOP. Two of these policies are broadly applicable to the class of valid utility games while the third is similarly applicable to valid utility games in extensive form.

To my family, who have always supported me.

ACKNOWLEDGMENTS

I would like to acknowledge all of the support provided over the past several years by my advisors, Professor Jugal Garg and Professor Rakesh Nagi, and thank them for it. I would also like to acknowledge the other members of my doctoral committee, Professor Carolyn Beck, Professor S. Rasoul Etesami, and Professor Rayadurgam Srikant.

Additionally, the work of myself and my advisors was supported through multiple grants. The work of Professor Garg was supported by NSF CRII under Award 1755619 while my work and that of Professor Nagi were supported in part by ONR through the Program Management of Drs. D. Wagner and W. Adams under Award N000014-16-1-2245.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

What causes suboptimal performance? Sometimes the cause is obvious and has a straightforward solution: Production is slow? Buy a new machine. A part breaks habitually? Get a better quality version or keep spares on hand. Something else unanticipated occurs? Invest in better forecasting and analyze the new data.

But sometimes even when the cause is obvious, the solution is not. Selfish behavior, the topic of this thesis, is one such cause. It appears in settings where two or more agents with non-aligned interests have to interact, and their selfish behavior frequently introduces inefficiency into any system they are acting in. Consider a classic example from Game Theory, the Prisoner's Dilemma. In this example, two thieves are caught attempting to rob a store and each is separately offered the same deal: Confess to everything and if your partner does not then you will go free and they will spend 20 years in prison, but if they also confess you will each go to prison for 5 years. Or, stay silent and if your partner does too then you can each go to prison for 1 year, but if your partner does confess you will go to prison for 20 years while they go free. Figure 1.1 illustrates the situation for each of the thieves.

If neither thief confesses then the pair will receive the minimum total amount of jail time, 2 years. However, no matter what one thief does, the other will benefit more by confessing: if one thief confesses their partner can receive 5 years if they also confess, or 20 if they remain silent. If one thief remains silent then the other can receive 1 year in prison if they also remain silent, or go free if they confess. Therefore, if both thieves are selfish then they will both confess and spend a total of 10 years in prison, rather than 2.

The importance of the effects of self-interested behavior is well-recognized and is the driving force behind much of the field of Game Theory. John Nash [1] showed the guaranteed existence of an equilibrium point in finite games, a point in which all agents have selected an entirely self-interested

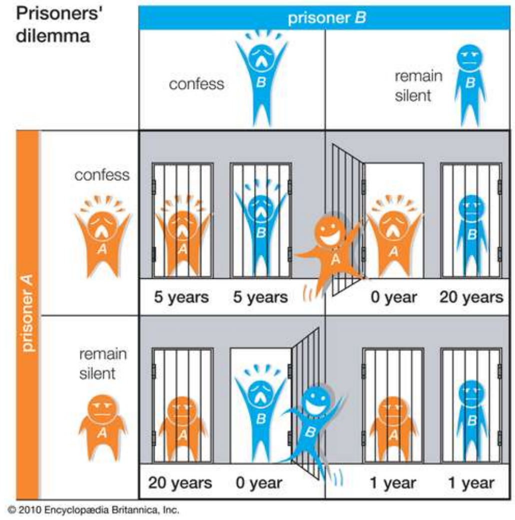

Figure 1.1: Sample Prisoner's Dilemma

Source: Encyclopedia Britannica, https://www.britannica.com/science/game-theory/The-prisoners-dilemma

strategy to the strategies selected by all other agents and thus, no agent has an incentive to change their strategy. This point has since been known as the *Nash equilibrium*, and is of great interest for predicting the actions of self-interested agents. In our sample prisoner's dilemma, the only Nash equilibrium occurs when both thieves confess, as doing so is the most self-interested action either can take.

With the assumption that agents will naturally arrive at a Nash equilibrium when behaving in a self-interested manner, a natural question is how inefficient could these equilibrium points be? In 2002 Adrian Vetta [2] showed that for the broad *Valid Utility* class of games, any Nash equilibrium point must achieve a minimum of 50% of the optimal outcome of the game. In Chapter 4 we define and analyze one such valid-utility game in extensive-form, the Prize-Collecting Multi-Agent Orienteering Problem, in order to design policies to reduce the amount of self-interested inefficiency in the problem with a limited degree of oversight. In doing so, we extend Vetta's result on valid utility games to the 2-player leader-follower setting. The policies proposed for the PCMOP are theoretically and numerically analyzed, and transplant naturally to other valid utility games, particularly those which are also in extensive-form.

The Nash equilibrium is a natural concept, and it explains interactions very well when agents are purely self-interested, as is often the case in a business or financial setting. However it provides fewer explanations when agents exhibit non-selfish behavior, or behavior which appears to be only partially self-interested. Because of this, before proposing and analyzing our policies for minimizing selfish behavior in the PCMOP and valid utility games in Chapter 4, in Chapter 2 we will address modeling and anticipating non-selfish behavior. This will be done by introducing the concept of Limited-Trust, in which one agent is willing to assist another agent provided the cost to the first agent is not too high. While superficially non-selfish, the first agent provides this aid with the intent of receiving reciprocal aid at some point in the future. Limited-trust forms the core of this manuscript, with Chapter 2 introducing, defining, and analyzing the concept, while Chapter 3 applies limited-trust to a social network setting. In doing so it demonstrates that despite performing "sub-optimally" for an agent in individual games, in aggregate the agent is far more likely to benefit than another agent playing solely in a self-interested manner. This is because by playing in a limited-trust manner, an agent is able to recapture some of the optimal utility which is lost due to selfish behavior. Therefore, while appearing selfless, this play-style actually acts as an enlightened form of selfishness for the agent.

The recapturing of the utility lost to selfish behavior by limited-trust provides a contrast to the policies considered in Chapter 4, which try to capture that same lost utility by limiting the ability of agents to be selfish. In particular, it demonstrates that for some settings with inefficient Nash-equilibria it is not necessary to implement policies to get rid of sub-optimal selfish behavior, as agents who are truly self-interested will eliminate these behaviors themselves to benefit in the long run. The importance of the work which will be covered in Chapters 2 and 3 is the quantification of how much sub-optimal behavior will remain in the system. With this knowledge, the manager of such a system can better determine what sort of policy is best to eliminate the remaining behavior, or whether it is even necessary to do so.

Finally, in Chapter 5 we will progress made since my preliminary exam in May 2020, as well as proposals for future research. Much of this will build on the social network games discussed in Chapter 3. However, we will also discuss fair-divisioning of chores as an extension of work in Chapter 4, with an emphasis on chores which are separated by transition times. Chapter 6 will then mark the end of this dissertation.

CHAPTER 2

LIMITED-TRUST EQUILIBRIUM: A NEW MODEL TO EXPLAIN NON-SELFISH BEHAVIOR

This chapter is based on work [3] which has been accepted to the *European Journal of Operational Research*.

2.1 Introduction

One of the first things children learn is to "play nice" with others. In order to get ahead and be a functioning member of society, each individual must sometimes make choices which do not appear to benefit them in the short term. Even though these actions cost the individual, they make up for it in benefit to society; over time each individual will have these costs returned to it in the form of unexpected favors. Under the rationality considered in a Nash equilibrium [1], it makes sense to pay these costs and only violate the social rules when the cost is too great. This is particularly true if, after a certain amount of time, the rule violation will be forgiven or forgotten. To that end, a great deal of research has gone into the study of extensive form games in general, and repeated games in particular. These games frequently evince equilibrium behaviors which, when only considered for the individual stages rather than the extensive game, are not rational under the Nash definition. An explicit discussion of the work on these games will be presented in Section 2.1.1.

However, when there are no formal consequences to avoiding the costs of society, such as exile, why do individuals continue to incur these costs? For example, why will most people give up a seat on a bus to a stranger who is injured? In a Nash equilibrium, in which only the utility of the individual making a decision is considered, the seat is never given up unless keeping it incurs some cost, such as damage to one's reputation. However, even if no one they know is present or will ever know of the decision, most people still give up the seat.

Perhaps the simplest answer is that the individual in the seat cannot know if their decision will ever make it back to others they interact with regularly and so they are simply risk averse. Another is that humans have some intrinsic degree of altruism. Evolutionary biology provides the best explanation of this in the form of Hamilton's rule for kin selection ([4–6]) which says that as humans are collections of genes, our genes seek to help any of the same genes present in other humans. To quote J. B. S. Haldane, "I would lay down my life for two brothers or eight cousins". This idea that we lend aid to others because they are some proportion of ourselves has given rise to the concept of α-altruism, which will be discussed in Section 2.1.1.

In this paper we introduce a new concept which we refer to as a limited-trust equilibrium. In it, a player i attempts to maximize its long-term utility by trusting the other player(s) within a hard trust limit δ_i that it is willing to give up when the other player(s) will gain "significantly" more than it (they) would lose if player i were to play "rationally". The player does this with the hope that the other player(s) will return the favor in a similar way, as well as form lasting partnerships and attract new ones through reputation. Given an opportunity, if an individual must choose between two agents of relatively equal capabilities to partner with then the individual would prefer to interact with the more trust-worthy agent. We show through numerical trials that in two player games, when both players have a similar trust limit, $\delta_1 = \delta_2 > 0$, both players come out significantly ahead in the long term compared to if they had played solely to maximize their own utility: in 2-player numerical trials with $\delta_1 = \delta_2 = \delta$ we observe an average personal utility increase of δ for each player when δ was modest compared to the value of the variance in the utilities of randomly generated games.

The limited-trust equilibrium provides a new answer to the previous question of why someone would give up their seat on the bus to an individual who is injured: they do so to establish and contribute to a culture of "kindness", which will increase the likelihood of someone giving them a seat in the event that they become ill or injured. This interpretation can be viewed as a person avoiding the consequences of the Broken Windows Theorem ([7]) which (loosely) states that evidence of erosion of one norm leads to further erosion of that and similar norms.

While it will be discussed more fully in Section 2.1.1, the idea of *non-*

rationality within repeated games has been extensively studied. Therefore we pause briefly to distinguish this concept from other solution concepts which occur within repeated games: in such situations, the same games are played repeatedly and so players arrive at a best way to handle that single game over time using methods such as future discounting and trigger strategies. In limited-trust games, while players are assumed to be playing with each other over time, they are not assumed to play the exact same game continuously. In fact, they may never play the same game twice. Because of this, it is necessary that one-off games be analyzed individually, as each game may be independent of previous or later games played. This is something that other tools for repeated game analysis cannot do. If two players do interact again, the game will most likely be different as it is assumed to be drawn from some probability distribution.

The rest of this paper is organized as follows: In Section 2.1.1 we provide a more detailed discussion of previous work into extended form games as well as α-altruism. In Section 2.2 we fully detail the properties of a limited-trust equilibrium (LTE): we show that it is guaranteed to exist in finite n-player games, prove where it fits within the hierarchy of equilibrium concepts (see Figure 2.1 for these results), and show that it results in higher net utility than Nash equilibria on several common games. Section 2.3 provides a mathematical program for LTE computation, and Section 2.4 discusses several interpretations of limited-trust in the leader-follower setting. In Section 2.5 we present the results of numerical trials in both the simultaneous and leader-follower settings, in which we compare the highest value Nash equilibria to the highest value LTE's for randomly generated games, before moving to our final discussion of results and concluding remarks in Section 2.6.

2.1.1 Literature Review

Since the seminal work of [1] there has been a great interest in Game Theory and equilibrium concepts. In particular, many papers have noted that the strict definition of rationality adhered to by Nash equilibria, that it is a state where no player can unilaterally improve its own utility given the actions of other players, is frequently *not* observed in empirical trials. One circumstance in which this occurs is repeated games in which players engage

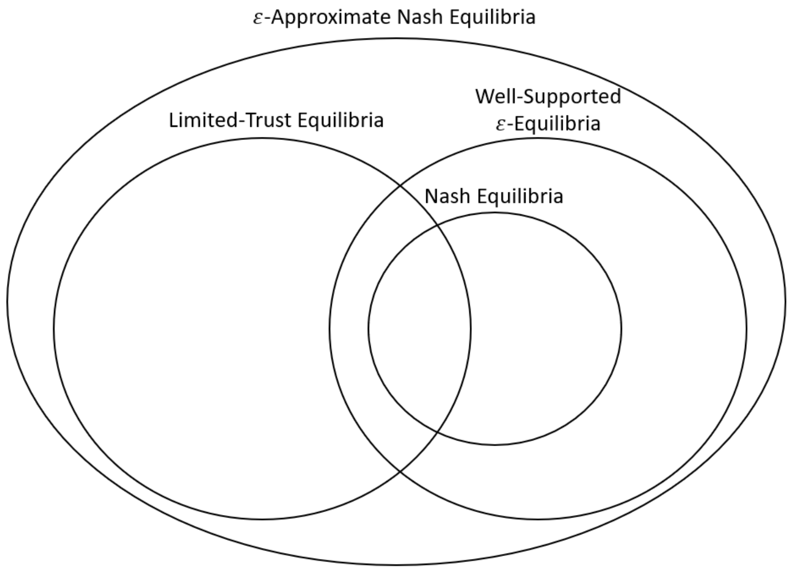

Figure 2.1: Heirarchy of equilibria. Intersection of all classes occurs in constant sum games.

in multiple rounds of play. Various *folk theorems* have been considered for these games which attempt to guarantee various measures of fairness in the equilibria; detailed analyses of these theorems and the conditions necessary for them to apply has been the subject of papers such as [8–11], and [12]. In the more applied sense, there has been a great deal of work aimed at developing rational definitions of trust for repeated games: papers such as [8, 13, 14] and [15] provide theoretical analysis of various games and trust strategies while papers such as [16], [17–20], and [21] have focused on conducting empirical studies on several of these trust strategies, particularly in the context of reciprocity. In the business setting [22] experimentally tests the real options games approach put forward by [23] for trust in strategic alliances. Meanwhile in the context of supply-chain relationships [24] empirically studies the formation of partnerships in the automotive industry, [25] derives a model for reciprocal-minded supplier-retailer relationships, and [26] and [27] empirically show supply-chain relationships tend to be more "fair" over time than predicted in standard game theory. The recent survey [28] details many of these as well as other empirical studies, all of which on average show *non-Nash* behavior.

These trust papers, both theoretical and experimental, deal explicitly with repeated games or (as in [18] and [16]) one-off games in extensive form (leader-follower). However, there is less work considering "non-rational" behavior in simultaneous one-off games. Most such work is done in the framework of α-altruism, as proposed by [29]. In this concept, each player i has a perceived utility of $u_i'(\sigma) = (1 - \alpha_i)u_i(\sigma) + \alpha_i u(\sigma)$ for $\alpha_i \in [0, 1]$ and thus takes the total social utility into account as part of its personal "utility". This model is attractive for a number of reasons: it is supported by Hamilton's kin-selection rule in evolutionary biology ([4–6]), it allows for easy equilibrium computation via Nash equilibria over perceived utilities, and it provides a broad model which can be adapted to virtually any form of game including simultaneous, extensive form, and repeated games. [30] provides a thorough analysis of this concept when applied to congestion, valid utility, and cost-sharing games, building on the analysis of [31] of this concept and extending the definition of (λ, μ)-smoothness put forth in an earlier version of [32] to α-altruistic games. However, this notion of altruism also has disadvantages, particularly from a modeling perspective. First, the game is scale invariant. This means that if player i would prefer not to collect €1 so that player j can collect an extra €2 given α_i, then it would prefer not to collect €100 so that player j can collect an extra €200 for the same α_i. Second, in games between a large number of players, the players are likely to become completely self-sacrificing to increase the total utility even for small $\alpha_i > 0$. To see this, consider a scenario in which for every unit of utility player i gives up, all other players receive some small amount of utility c, where $0 < c \ll 1$. As the number of players grows, player i will seek to drive its personal utility as low as possible so long as $\alpha_i > 0$.

In the next section we will propose a new concept of a limited-trust equilibrium which applies to a similarly broad class of games, but incorporates a hard trust-limit not present in α-altruism. Players behave in a manner which encourages reciprocity, provided it is not to expensive for them personally in terms of a hard limit on their current personal utility. They make this investment in reciprocation in order to increase their personal utilities in the long run or in expectation. This concept places a "budget" on what players spend toward encouraging reciprocity in any one game and thus eliminates both the tendency of players in large games to become self-sacrificing and the scale invariance which occur in α-altruistic games.

2.2 Limited-Trust Equilibrium

We now define a new concept of equilibrium in which players, while still selfish and concerned primarily with their own utility, exhibit a limited interest in the common good and contribute to it provided the cost is below some threshold. They do so in order to encourage other players to do the same in order to benefit in the long term. For comparison, we first review the definition of a mixed Nash equilibrium (MNE) over a finite game:

Definition 1 (Strategy Profile of a Finite Game)**.** *Given a finite n-player game in which each player i has m_i pure strategies, a valid strategy profile σ_i for player i is a probability distribution over the m_i pure strategies ($\sigma_i = \{p^i_1, p^i_2, ..., p^i_{m_i}\}$, $p^i_j \geq 0$, $\sum_{j=1}^{m_i} p^i_j = 1$).*

Definition 2 (Mixed Nash Equilibrium)**.** *Given an n-player game with strategy profiles $\sigma = (\sigma_1, \sigma_2, ..., \sigma_n)$ for each player where for a given player i, σ_{-i} is the set of strategies played by all other players, σ is a mixed Nash equilibrium (MNE) if and only if for any other valid strategy profiles σ'_i, $u_i(\sigma_i, \sigma_{-i}) \geq u_i(\sigma'_i, \sigma_{-i})$ for all $i \in [n]$, where $[n] = \{1, 2, ..., n\}$ and $u_i(\sigma_i, \sigma_{-i})$ is the expected utility of the game for player i.*

A related concept is the ε-approximate Nash equilibrium (ε-equilibrium) defined as follows:

Definition 3 (ε-Approximate Nash Equilibrium)**.** *For an n-player game with strategy profiles $\sigma = (\sigma_1, \sigma_2, ..., \sigma_n)$ for each player, σ is an ε-equilibrium if and only if for any other valid strategy profiles σ'_i, $u_i(\sigma_i, \sigma_{-i}) \geq u_i(\sigma'_i, \sigma_{-i}) - \varepsilon$ for all $i \in [n]$.*

Definition 4 (Price of Anarchy)**.** *The Price of Anarchy (PoA) of a utility maximization game is the ratio of the the value of the socially optimal solution, defined as the solution that maximizes the sum of the utilities of all players (net utility), to the value of the equilibrium with the lowest social utility.*

Typically the equilibrium considered in the PoA is the Nash equilibrium; in this paper we will explicitly state which equilibrium is being considered when using the term.

Note that the set of Nash equilibria is merely the set of ε-approximate equilibria for $\varepsilon = 0$. It is also worth noting that the conditions of an MNE can be defined in mathematical constraints. For an n-player utility maximization game, any strategy profile σ comprises an MNE if and only if it satisfies the following constraints:

$$u_i(\sigma_i', \sigma_{-i}) - u_i(\sigma_i, \sigma_{-i}) \leq 0 \qquad \forall \sigma_i' \in \Sigma_i,\ i \in [n]$$

where Σ_i is the set of valid strategy profiles for player i. We also define

$$\sigma_i^G(\sigma_{-i}) = \arg\max_{\sigma_i} u_i(\sigma_i, \sigma_{-i})$$

as the greedy best response of player i given σ_{-i}. We will abuse notation to let $\sigma_i^G \in \sigma_i^G(\sigma_{-i})$; while there may be multiple elements of $\sigma_i^G(\sigma_{-i})$, as it is a set-valued function, we will only be concerned with σ_i^G with regard to the value $u_i(\sigma_i^G, \sigma_{-i})$ which is equal for all elements of $\sigma_i^G(\sigma_{-i})$. We say that an ε-equilibrium $\sigma \in \Sigma$ is *well-supported* if and only if for every player i, $u_i(\sigma_i^G, \sigma_{-i}) - u_i(s_j^i, \sigma_{-i}) \leq \varepsilon$ for every pure strategy s_j^i which is played in σ_i with non-zero probability. Note that any MNE is a well-supported ε-equilibrium for all $\varepsilon \geq 0$.

Having covered our preliminary definitions, we now propose a new concept of equilibrium.

Definition 5 (Limited-Trust Equilibrium (LTE))**.** *Consider a finite n-player maximization game with strategy profiles $\sigma \in \Sigma = \Sigma_1 \times ... \times \Sigma_n$ and trust levels $\delta = (\delta_1, ..., \delta_n)$ for each player i, where $\delta_i > 0$. σ is a limited-trust equilibrium if and only if $u_i(\sigma_i^G, \sigma_{-i}) - u_i(\sigma_i, \sigma_{-i}) \leq \delta_i$ and $u(\sigma_i, \sigma_{-i}) \geq u(\sigma_i', \sigma_{-i})$ for any other valid strategy profiles $\sigma_i' \in \Sigma_i$ such that $u_i(\sigma_i^G, \sigma_{-i}) - u_i(\sigma_i', \sigma_{-i}) \leq \delta_i$, where $u(\sigma) = \sum_{i=1}^n u_i(\sigma)$ is the net utility.*

We will use LTE(δ) to refer to an LTE for players with trust levels $\delta = \{\delta_1, ..., \delta_n\}$. This definition is equivalent to saying that the following two conditions are met:

1. Player i cannot alter its strategy profile to increase its payoff by more than δ_i. In other words, it is not giving up more than δ_i it could be making by changing its behavior to take advantage of other players' strategies.

2. Player i cannot alter its strategy profile to increase the net utility without decreasing its own utility so that it loses more than δ_i from its

greedy best response. In other words, it cannot increase the net utility without violating its cost threshold δ_i.

As for where δ comes from, it can be viewed as the degree to which an individual is willing to invest in the future, meaning the cost they are willing to incur in order to benefit others and encourage them to reciprocate.

It is worth noting that we could equivalently write the net utility $u(\sigma_i, \sigma_{-i})$ as

$$u(\sigma_i, \sigma_{-i}) = \sum_{i=1}^{n} u_i(\sigma_i, \sigma_{-i}) = u_i(\sigma_i, \sigma_{-i}) + u_{-i}(\sigma_i, \sigma_{-i})$$

where $u_{-i}(\sigma_i, \sigma_{-i}) = \sum_{j \neq i} u_j(\sigma_i, \sigma_{-i})$. While there is no mathematical advantage in doing so, it helps to illustrate that if player i gives up δ_i by playing σ'_i rather than σ_i and the net utility increases by $x < \delta_i$, the $\delta_i - x$ value is not simply lost. Rather, if $u_i(\sigma'_i, \sigma_{-i}) - u_i(\sigma_i, \sigma_{-i}) = -\delta_i$ this means $u_{-i}(\sigma'_i, \sigma_{-i}) - u_{-i}(\sigma_i, \sigma_{-i}) = \delta_i + x$.

Because a limited-trust (LT) best response is concerned with two values $u(\sigma)$ and $u_i(\sigma)$ it makes sense to examine their relationship. In particular for a player i, if all other players are playing σ_{-i} then player i can easily determine the results of all of its pure strategies s^i_j in terms of $u(\sigma_i, \sigma_{-i})$ and $u_i(\sigma_i, \sigma_{-i})$. Because of the linearity of u and u_i with respect to s^j_i given a fixed σ_{-i}, any u, u_i combination within the convex hull of the pure strategies can be achieved by player i. Therefore, player i can solve the following linear program LP1 to find its limited-trust best response $\sigma^*_i(\sigma_{-i})$:

$$\begin{aligned} &\sigma^*_i(\sigma_{-i}) = \arg\max_{\sigma_i \in \Sigma_i} u(\sigma_i, \sigma_{-i}) \\ \text{subject to} \\ &\delta_i \geq u_i(\sigma^G_i, \sigma_{-i}) - u_i(\sigma_i, \sigma_{-i}). \end{aligned} \tag{LP1}$$

When we take the limit $\delta_i \to \infty$ we find that player i becomes completely self-sacrificing for the net utility. Thus by careful selection of δ, players of any degree of trustworthiness from completely self-interested ($\delta \to 0$) to completely selfless ($\delta \to \infty$) may be modeled, though players of the latter type may be quite uncommon.

We note that while the set $\sigma^G(\sigma) = \{\sigma^G_1(\sigma_{-1}), \sigma^G_2(\sigma_{-2}), \ldots, \sigma^G_n(\sigma_{-n})\}$ is the set of greedy best responses to the current strategy set σ, $\sigma^G \in \sigma^G(\sigma)$ is not generally a Nash equilibrium. Instead, it is merely a set of greedy best responses to σ for each player. As such, the fact that σ_i^G is a component of player i's limited-trust best response does not imply that a limited-trust equilibrium is dependent on a Nash equilibrium, merely that is dependent on greedy best responses. As Nash equilibria are also heavily dependent on greedy best responses, with $\sigma \in \sigma^G(\sigma)$ being a necessary and sufficient condition for σ to be a Nash equilibrium, this can be a subtle point. To further emphasize this distinction, we show in an example game in Table 2.4 that has an LTE can exist independent of any Nash equilibrium.

Lemma 1. *Given an LTE(δ) σ^*, if a constant c_j is added to all payoffs for player j, σ^* is still an LTE(δ).*

The proof is included in A.1.1.

Although Lemma 1 demonstrates that an LTE(δ) is invariant under the addition of a constant c_j to all of player j's payoffs, the same is not true for affine transformations. This is an intentional feature of the limited-trust concept: while a player may be willing to accept a loss of €1 to ensure another player gains €2, it is not willing to accept a loss of €100 to ensure another player gains €200, as would be required of an affine transformation of a game. However, for a given affine transformation $f(x) = ax + b$ the equilibria are invariant if δ is rescaled to $\delta|a|$, for $a \neq 0$.

Theorem 1. *Every n-player finite game with trust levels $\delta = (\delta_1, \delta_2, ..., \delta_k) > 0$ has an LTE(δ).*

Proof. This proof will follow the same pattern as Nash's ([1]) proof for the existence of MNE in an n player game by making use of Kakutani's Fixed Point Theorem ([33]).

To begin, let $\sigma \in \Sigma$ be a set of strategy profiles for each player. Let $u_i(\sigma) = u_i(\sigma_i, \sigma_{-i})$ be the payoff player i derives from strategy profile σ_i given that all other players are playing σ_{-i}. Now we wish to define a new utility function

$$w_i(\sigma_i, \sigma_{-i}) = \begin{cases} u(\sigma_i, \sigma_{-i}) & u_i(\sigma_i^G, \sigma_{-i}) - u_i(\sigma_i, \sigma_{-i}) \leq \delta_i \\ -M & \text{otherwise,} \end{cases}$$

where M is a large positive constant. Because the game is finite, we can pick M greater than maximum of the absolute values of the socially optimal solution and the most socially harmful solution multiplied by n, and the $w_i(\sigma_i, \sigma_{-i})$ of any σ_i which violates $u_i(\sigma_i^G, \sigma_{-i}) - u_i(\sigma_i, \sigma_{-i}) \le \delta_i$ is strictly less than $w_i(\sigma_i', \sigma_{-i})$ for some σ_i' which does not. Therefore, maximizing $w_i(\sigma_i, \sigma_{-i})$ is equivalent to maximizing $u(\sigma_i, \sigma_{-i})$ over the set of points which satisfy the maximum cost constraint. We can then say that $\sigma^* \in \Sigma$ is a LTE(δ) if and only if

$$w_i(\sigma_i^*, \sigma_{-i}^*) \ge w_i(\sigma_i, \sigma_{-i}^*) \qquad \forall \sigma_i \in \Sigma_i,\ \forall i \in \{1, 2, ..., n\},$$

which means that σ^* is a LTE(δ) if and only if $\sigma_i^* \in B_i(\sigma_{-i}^*)$ for all i, where $B_i(\sigma_{-i}^*)$ is the set of best responses (with respect to w_i) for player i given that the other players are playing σ_{-i}^*. If we define $B(\sigma) = B_1(\sigma_{-1}) \times B_2(\sigma_{-2}) \times ... \times B_n(\sigma_{-n})$ then finding an LTE(δ) is equivalent to finding $\sigma \in B(\sigma)$. Therefore, we must show the existence of a fixed point.

We now use Kakutani's Fixed Point Theorem to show such a fixed point exists. The theorem states that given a nonempty finite dimensional Euclidean space A and $f : A \to A$ a set-valued correspondence with $x \in A \to f(x) \subseteq A$, a fixed point is guaranteed to exist if the following conditions hold:

1. A is a compact and convex set.

2. $f(x)$ is nonempty for all $x \in A$.

3. $f(x)$ is convex for all $x \in A$.

4. $f(x)$ has a closed graph: if $\{x^k, y^k\} \to \{x, y\}$ with $y^k \in f(x^k)$ then $y \in f(x)$.

In this case we have $A = \Sigma$, $f(\sigma) = B(\sigma)$. We now wish to show that all conditions hold.

1. Σ is a compact and convex set: trivial, as Σ is the Cartesian product of simplices Σ_i.

2. $B(\sigma)$ is nonempty for all $\sigma \in \Sigma$: $B_i(\sigma_{-i}) = \arg\max_{\sigma_i \in \Sigma_i} w_i(\sigma_i, \sigma_{-i})$ and so must be nonempty for each i. Therefore $B(\sigma)$ is nonempty for all $\sigma \in \Sigma$.

3. $B(\sigma)$ is convex for all $\sigma \in \Sigma$: It suffices to show that $B_i(\sigma_{-i})$ is convex for all i. We first note that any points $x, y \in B_i(\sigma_{-i})$ must provide

equal net utility $u(x, \sigma_{-i}) = u(y, \sigma_{-i})$ and must also provide i with a personal utility at most δ_i less than the greedy best response. Without loss of generality, assume $u_i(x, \sigma_{-i}) \geq u_i(y, \sigma_{-i})$. Then for any convex combination $z = \lambda x + (1 - \lambda)y$ where $\lambda \in [0, 1]$, the linearity of u and u_i implies that $u(x, \sigma_{-i}) = u(z, \sigma_{-i}) = u(y, \sigma_{-i})$ and $u_i(x, \sigma_{-i}) \geq u_i(z, \sigma_{-i}) \geq u_i(y, \sigma_{-i})$ which means $z \in B_i(\sigma_{-i})$.

4. $B(\sigma)$ has a closed graph: While the previous three conditions were shown to hold using the same arguments as in the proof of existence for Nash equilibria, the use of a non-continuous function w_i introduces several complications to showing that $B(\sigma)$ has a closed graph. We will show this by contradiction. Suppose that $B(\sigma)$ does not have a closed graph. Then there exists a sequence $(\sigma^k, \hat{\sigma}^k) \to (\sigma, \hat{\sigma})$ such that $\hat{\sigma}^k \in B(\sigma^k)$, but $\hat{\sigma} \notin B(\sigma)$, meaning that $\hat{\sigma}_i \notin B_i(\sigma_{-i})$ for some i. Then there is some $\sigma'_i \in B(\sigma_{-i})$ such that

$$w_i(\sigma'_i, \sigma_{-i}) > w_i(\hat{\sigma}_i, \sigma_{-i}),$$

which means that

$$u(\sigma'_i, \sigma_{-i}) > u(\hat{\sigma}_i, \sigma_{-i}).$$

By continuity of u_i and u, we have that for k sufficiently large $u(\sigma'_i, \sigma^k_{-i}) > u(\hat{\sigma}^k_i, \sigma^k_{-i})$. Because $\sigma'_i \notin B_i(\sigma^k_{-i})$, we have $u_i(\sigma^{Gk}_i, \sigma^k_{-i}) - u_i(\sigma'_i, \sigma^k_{-i}) > \delta_i$ where $\sigma^{Gk}_i \in \sigma^G_i(\sigma^k_{-i})$ as otherwise this would contradict the assumption that $\hat{\sigma}^k_i \in B_i(\sigma^k_{-i})$.

Suppose that $u_i(\sigma^{Gk}_i, \sigma^k_{-i}) - u_i(\hat{\sigma}^k_i, \sigma^k_{-i}) < \delta_i$. Then by the linearity of u_i and u there is a convex combination s of σ'_i and $\hat{\sigma}^k_i$ such that $u_i(\sigma^{Gk}_i, \sigma^k_{-i}) - u_i(s, \sigma^k_{-i}) \leq \delta_i$ and $u(s, \sigma^k_{-i}) > u(\hat{\sigma}^k_i, \sigma^k_{-i})$, which implies $w_i(s, \sigma^k_{-i}) > w_i(\hat{\sigma}^k_i, \sigma^k_{-i})$ and therefore contradicts the assumption that $\hat{\sigma}^k_i \in B_i(\sigma^k_{-i})$.

Now suppose that $u_i(\sigma^{Gk}_i, \sigma^k_{-i}) - u_i(\hat{\sigma}^k_i, \sigma^k_{-i}) = \delta_i$. In order for σ'_i to become a strategy in $B_i(\sigma_{-i})$, it must be that $u_i(\sigma^G_i, \sigma_{-i}) - u_i(\sigma'_i, \sigma_{-i}) = \delta_i$, as if it became less than or equal to δ_i for sufficiently high k, then it would contradict $\hat{\sigma}^k_i \in B_i(\sigma^k_{-i})$. Similarly, $u_i(\sigma^G_i, \sigma_{-i}) - u_i(\hat{\sigma}_i, \sigma_{-i}) = \delta_i$ as if it is less than δ_i, then for sufficiently high k we would have $u_i(\sigma^{Gk}_i, \sigma^k_{-i}) - u_i(\hat{\sigma}^k_i, \sigma^k_{-i}) < \delta_i$ which we have already seen leads to a contradiction. However, this means that there is a strategy $\sigma_i'' = \lambda\sigma_i^G + (1-\lambda)\sigma_i'$ for some value of $\lambda \in [0,1]$ which has $u(\sigma_i'', \sigma_{-i}) > u(\hat{\sigma}_i, \sigma_{-i})$ and $u_i(\sigma_i'', \sigma_{-i}) > u_i(\hat{\sigma}_i, \sigma_{-i})$, due to the assumption that $\delta_i > 0$ and the linearity of u and u_i. This means that $u(\sigma_i'', \sigma_{-i}^k) > u(\hat{\sigma}_i^k, \sigma_{-i}^k)$ and $u_i(\sigma_i'', \sigma_{-i}^k) > u_i(\hat{\sigma}_i^k, \sigma_{-i}^k)$ for sufficiently high k. Given $u_i(\sigma_i^{Gk}, \sigma_{-i}^k) - u_i(\hat{\sigma}_i^k, \sigma_{-i}^k) \le \delta_i$, this means $w_i(\sigma_i'', \sigma_{-i}^k) > w_i(\hat{\sigma}_i^k, \sigma_{-i}^k)$ which contradicts the assumption that $\hat{\sigma}_i^k \in B(\sigma_{-i}^k)$. Therefore $B(\sigma)$ must have a closed graph.

Therefore, Kakutani's theorem implies the existence of a $\sigma^* \in \Sigma$ such that $\sigma^* \in B(\sigma^*)$, which proves the existence of an LTE(δ). □

Having established the guaranteed existence of an LTE for $\delta > 0$ we now want to compare it to a Nash equilibrium on a simple example, given in Table 2.1. This game has exactly one Nash equilibrium, at $\sigma = \{[0,1],[0,1]\}$ with pure strategies α_2, β_2 being played. Now consider the LTE with $\delta = \{0.5, 0.5\}$. LP1 shows that for player 2, playing the pure strategy β_2 ($\sigma_2 = [0,1]$) is still the best choice, regardless of σ_1. The same is not true for player 1: given $\sigma_2 = [0,1]$, solving LP1 gives the first player's unique best response as $\sigma_1 = [0.5, 0.5]$. The net utility of the LTE(0.5) for the game is then $u_1(\{[0.5,0.5],[0,1]\}) + u_2(\{[0.5,0.5],[0,1]\}) = 9$, compared to the net utility of 8 which occurs in the Nash equilibrium.

Table 2.1: Example game for LTE

		Player 2	
		β_1	β_2
Player 1	α_1	4,0	5,5
	α_2	5,1	6,2

The equilibrium in Table 2.1 highlights an important fact about the limited-trust best response, that there may not be a pure strategy best response. This is at odds with the greedy best response where there is always a pure strategy best response. This implies that there does not appear to be a straightforward transformation of a limited-trust game into a Nash game.

Next, we wish to consider where the LTE fits within the hierarchy of standard solution concepts within game theory.

Theorem 2. *For any finite n-player game G, the set of LTE(δ) is a subset of the set of ε-equilibria of G, where $\varepsilon \ge \max_i \delta_i$.*

Proof. Consider that in any LTE(δ), no player can improve its own payoff by more than δ_i by definition of an LTE. Therefore such an LTE is also an ε-equilibrium for $\varepsilon \geq \max_i \delta_i$. $\square$

While each limited trust equilibrium is also a $\max_i \delta_i = \varepsilon$-equilibrium, the converse is not true, even when $\delta_i = \delta_j$ for all $\forall i, j \in [n]$. This is because of the additional constraint on an LTE(δ) that no player i be able to improve the total utility without decreasing its own utility below the δ_i level. Further, although the set of LTE(δ)'s is a subset of ε-equilibria as described in Theorem 2, they are important because they represent a state in which each player is contributing to the net utility as much as they are able within their limits, not merely a state where each player has decided it is not worth the effort (or in the case of irrational-valued Nash equilibria it is realistically infeasible) to change from their current strategy to the optimal strategy, particularly if the current strategy is pure.

In general, we say that an LTE(δ) σ is well-supported if it is a well-supported ε-equilibrium for $\varepsilon = \max_i \delta_i$. Although any LTE($\delta$) is an ε-equilibrium for ε as previously specified, it need not be a well-supported ε-equilibrium . The 2-player game in Table 2.2 demonstrates this for $\delta_1 = \delta_2 = 0.5$. From player 2's limited-trust perspective, β_2 is a best response to any σ_1 as it offers both better personal and better net utility. Player 1's limited-trust best response to β_2 is to play α_i with probability 0.5, for $i = \{1, 2\}$. Given that player 2 will only play β_2, the only LTE($[.5, .5]$) is given by $\{[0.5, 0.5], [0, 1]\}$. This is not a well-supported 0.5-equilibrium: $\sigma_1^G = \alpha_2$ and player 1 is playing α_1 with nonzero probability, despite the fact that $u_i(\sigma_i^G, \beta_2) - u_i(\alpha_1, \beta_2) = 1 > 0.5$. As this is the only LTE($[0.5, 0.5]$), this game also shows that the set of LTE(δ) may be entirely disjoint from the set of well-supported ε-equilibria for a game.

Table 2.2: Game with non-overlapping LTE($[0.5, 0.5]$) and well-supported 0.5-equilibria

		Player 2: β_1	Player 2: β_2
Player 1	α_1	2,4	3,5
	α_2	3,2	4,3

Despite the fact that players in a limited-trust game all attempt to improve the net utility, it is possible for the highest value LTE(δ) (the LTE(δ) which

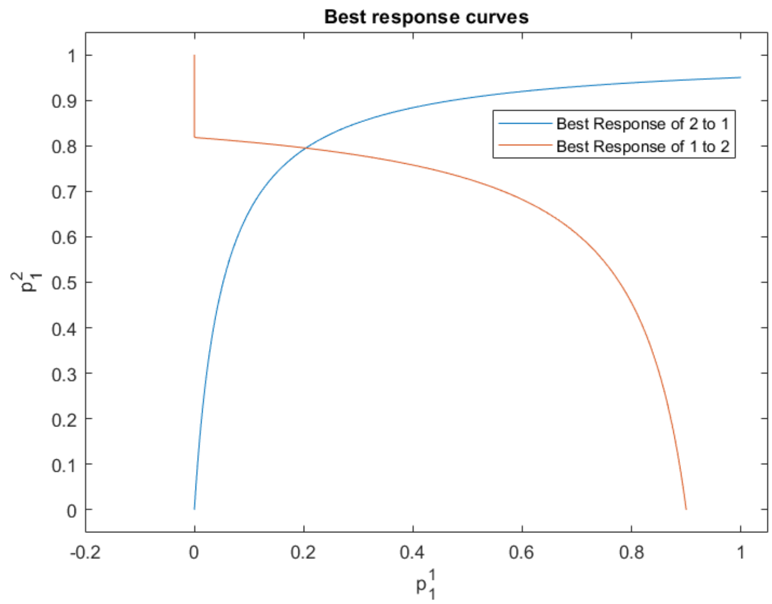

Figure 2.2: Best response curves for game in Table 2.3, $\delta_1 = \delta_2 = 0.1$

provides the highest net utility) to produce lower net utility than the highest value Nash equilibrium for a game. We show this with an example game, given in Table 2.3, which we consider with $\delta_1 = \delta_2 = 0.1$. There is a pure Nash equilibrium (PNE) which occurs for α_2, β_1. As β_1 is a strongly dominant strategy for player 2, and α_2 is player 1's best response to it, this is the only Nash equilibrium. If we consider the limited-trust best response curves in Figure 2.2, we see that there is only one place the curves intersect and hence there is one LTE(0.1). Using optimization program MP1 from Section 2.3, we find this is at approximately $\sigma_1 = \{p_1^1, p_2^1\} = \{0.204, 0.796\}$, $\sigma_2 = \{p_1^2, p_2^2\} = \{.795, .205\}$ which has a total utility value of approximately 6.089, which is less than 6.2, the utility generated by the pure Nash equilibrium at $\sigma_1 = \{0, 1\}$, $\sigma_2 = \{1, 0\}$.

Table 2.3: Game in which LTE(0.1) < MNE

		Player 2	
		β_1	β_2
Player 1	α_1	1,2	5,0
	α_2	1.1,5.1	4,5

While it is non-intuitive that the value of the best LTE can be less than a Nash equilibrium, given that each player is willing to give something up in order to help its fellow players, we do see analogues of this in the day-to-day social interactions which the concept of limited-trust emulates. Consider two cars reaching an intersection across from each other. Both need to turn left and the intersection is too narrow for both to go at once. Rather than attempting to go through first, one driver tries to wave the other through, only to realize that the other driver is doing the same. Both drivers start to move, then stop as they realize the other is moving as well. This then repeats back and forth until one driver loses their patience (δ is reached) and makes it clear they are going. Meanwhile the whole interaction slowed down both drivers more than if one had simply made this decision when they both arrived at the intersection.

Although it is possible to find games in which there is a Nash equilibrium better than any LTE, we will see in Section 2.5 that it rarely occurs, particularly as δ increases; it is more common to find games in which there are more LTE(δ) than Nash equilibria and some of them are worse. Table 2.4 provides an example of this, where (α_2, β_2) is both an LTE(δ) and a pure Nash equilibrium. However, for $\delta_1, \delta_2 \geq 0.1$, (α_1, β_1) is also an LTE(δ), independent of a Nash equilibrium.

Table 2.4: Game with more LTE(δ) than Nash equilibria

		Player 2	
		β_1	β_2
Player 1	α_1	3,3	2,3.1
	α_2	3.1,2	5,5

Further, the occurrence of less optimal solutions due to cooperation is not unique to limited-trust games: [30] shows that while normal cost-sharing games have a PoA of n for n players, cost-sharing games in which all players have a uniform level of α-altruism have a PoA of $\frac{n}{1-\alpha}$, becoming unboundedly inefficient for fully altruistic players. The remainder of this section will be spent considering limited-trust versions of several standard games, and we will see that Theorem these inefficiencies do not apply to them.

2.2.1 LTE(δ) in Common Games

In this section we examine the behavior and value of LTE(δ) in several common classes of games.

Theorem 3. *For any constant sum game, {ε-equilibria} $\subseteq$ {LTE(δ)} where* $\varepsilon = \min_i \delta_i$.

Proof. First, note that in a constant sum game the total utility is equal for all $\sigma \in \Sigma$. Therefore, any strategy σ played by player i maximizes the total utility. Thus, player i's best response to any σ_{-i} is any strategy which makes sure it receives at most δ_i less than its maximum personal utility. This is exactly the definition of a best response under ε-equilibrium conditions for $\delta_i = \varepsilon$, and so the set of LTE contains the set of ε-equilibria for a constant sum game where $\varepsilon = \min_i \delta_i$. □

Note that Theorem 2 states the LTE set is a subset of the ε-equilibria set for $\varepsilon = \max_i \delta_i$, so if $\delta_i = \delta_j = \delta$ for all $i \neq j$ then Theorem 3 implies the set of ε-equilibria is equal to the set of LTE(δ) for $\varepsilon = \delta$.

Next we consider the public goods game from experimental economics. In it, n players each receive an amount of money, m_i, and must decide how much to contribute to the public good. Any contributed money is multiplied by a factor of c such that $1 < c < n$, then divided evenly among all players. Therefore, if player i contributes x_i to the public good, it will receive back $\frac{cx_i}{n} < x_i$ of its investment, plus $\frac{c}{n}$ of the other players' investments. The only Nash equilibrium for this game is for all players to contribute $x_i = 0$, as any contribution lowers player i's payoff regardless of contributions made by other players. The PoA is therefore c.

Theorem 4. *In a public goods game with δ, the limited-trust PoA is*

$$\frac{c\sum_{i=1}^{n} m_i}{\sum_{i=1}^{n} m_i + (c-1)\min\{\frac{n}{n-c}\delta_i, m_i\}} \leq c.$$

Proof. Consider the contribution player i should make: the social utility strictly increases with i's contribution x_i, therefore player i would like to contribute as much as possible. i is willing to lose at most δ_i and regardless of the value of x_j for $j \neq i$, if player i contributes x_i then it loses $\frac{n-c}{n}x_i$ it could be making. Therefore, player i contributes $x_i = \min\{\frac{n}{n-c}\delta_i, m_i\}$. The total amount contributed is $\sum_{i=1}^{n} x_i$, and the total uncontributed utility is $\sum_{i=1}^{n} m_i - x_i$, which means that the total utility generated is $\sum_{i=1}^{n} m_i + (c-1)\min\{\frac{n}{n-c}\delta_i, m_i\}$. This is the unique LTE(δ) for the public goods game. The socially optimal result occurs when all players contribute m_i and there is a

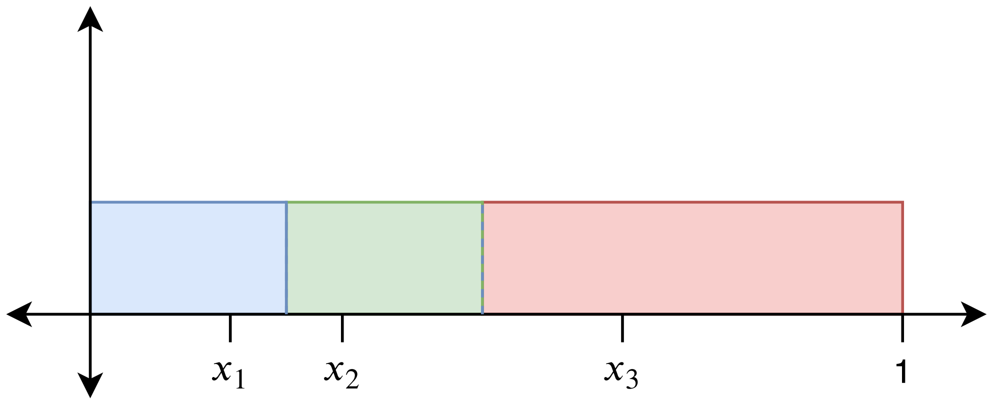

Figure 2.3: Hotelling Game with $n = 3$ players

total utility of $c\sum_{i=1}^{n} m_i$, so the limited-trust PoA is $\frac{c\sum_{i=1}^{n} m_i}{\sum_{i=1}^{n} m_i + (c-1)\min\{\frac{n}{n-c}\delta_i, m_i\}}$ that is at most c. $\square$

Our next consideration is the Hotelling game, which does not generally have a pure Nash equilibrium for any number of players n We will consider the simplest form of the game, in which each player has a continuous strategy space $[0, 1]$ and all players have symmetric payoffs, meaning that for any two players i, j and all other strategies σ_{-ij} fixed, if i and j switched strategies they would also switch utilities ($u_i(\sigma_i = x_1, \sigma_j = x_2, \sigma_{-ij}) = u_j(\sigma_i = x_2, \sigma_j = x_1, \sigma_{-ij})$ for all $x_1, x_2 \in [0, 1]$). Given strategies $\{\sigma_1, \sigma_2, ..., \sigma_n\}$, if we assume without loss of generality that $0 \leq \sigma_1 \leq \sigma_2 \leq ... \leq \sigma_n \leq 1$ then for $\sigma_{i-1} < \sigma_i < \sigma_{i+1}$ $u_i(\sigma) = \frac{\sigma_{i+1} - \sigma_{i-1}}{2}$. If there is a set of k strategies $\sigma_i = \sigma_{i+1} = ... = \sigma_{i+k-1}$, then $u_j(\sigma) = \frac{\sigma_{i+k} - \sigma_{i-1}}{2k}, \forall i \leq j \leq i + k - 1$. Additionally, for the purposes of computing σ_1 and σ_n, let "σ_0" $= -\sigma_1$ and "σ_{n+1}" $= 1 + \sigma_n$.

This simple form of the Hotelling game can be viewed as each player claiming a space on the interval $[0, 1]$, with each player attempting to maximize the portion of the interval which is closer to them then all other players. Figure 2.3 provides an example of this for a 3-player Hotelling game, which does not have a pure Nash equilibrium.

Theorem 5. *The 3-player Hotelling game possesses a pure LTE(δ) for $\delta_i \geq \frac{1}{10}$ for $i \in \{1, 2, 3\}$.*

Proof. This will be a proof by example, showing that $\sigma = \{\frac{3}{10}, \frac{1}{2}, \frac{7}{10}\}$ is an LTE(δ) for $\delta_i \geq \frac{1}{10}$ for $i \in \{1, 2, 3\}$. We begin by noting that the Hotelling game is constant-sum, so any strategy produces the same net utility. We first consider whether player 1 is at equilibrium. Observe that $u_1(\sigma) = \frac{2}{5}$, so player 1 is at equilibrium provided there is not some σ_1' such that $u_1(\sigma_1', \sigma_{-1}) > \frac{1}{2}$. As $\sigma_2 = \frac{1}{2}$, there cannot be: $\sigma_1' \in [0, \frac{1}{2}), (\frac{1}{2}, 1]$ will result in utility strictly less than $\frac{1}{2}$, and $\sigma_1' = \frac{1}{2}$ will result in the same utility as player 2 receives, which can be at most $\frac{1}{2}$ as the net utility for the game is 1. Therefore player 1 is at equilibrium, and similarly player 3 is at equilibrium as its position is symmetric to that of player 1.

This leaves player 2. $u_2(\sigma) = \frac{1}{5}$, so it is at equilibrium if there is no σ_2' such that $u_2(\sigma_2', \sigma_{-2}) > \frac{3}{10}$. For $\sigma_2' \in [0, \frac{3}{10}), (\frac{3}{10}, \frac{7}{10}), (\frac{7}{10}, 1]$, $u_2(\sigma_2', \sigma_{-2})$ is $< \frac{3}{10}, \frac{1}{5}$, and $< \frac{3}{10}$ respectively. For $\sigma_2' = \frac{3}{10}, \frac{7}{10}$ $u_2(\sigma_2, \sigma_{-2}) = \frac{1}{4} < \frac{3}{10}$ as well, so player 2 is at equilibrium. $\square$

Before moving on, it is worthwhile to note that although Theorem 1 only implies the existence of the LTE(δ) in finite normal form games, players in both the public goods game and Hotelling game possess a continuous rather than finite strategy set. This helps to highlight that while non-finite games are outside the scope of this paper, many classes of these games are also likely to possess limited-trust equilibria.

Finally, we consider the 2×2 prisoner's dilemma, though we will focus on the utility maximization version rather than the cost minimization version. Let (α_2, β_2) be the socially optimal outcome, and let (α_1, β_1) be the strategy in which each player betrays the other. In a Nash game, the only equilibrium is (α_1, β_1), the worst possible outcome. In the limited-trust game, $\lim_{\delta \to 0}$ LTE(δ) is (α_1, β_1), but as δ increases, it shifts to (α_2, β_2). Table 2.5 shows the general form of a symmetric version of the game, with $0 < d_1 < c < d_2$. By noting the fact that both players will be playing the same strategy $\sigma = \{p_1, 1 - p_1\}$ at equilibrium if $\delta_1 = \delta_2$, we can find the LTE(δ) by solving the quadratic equation $(1 - p_1)(p_1 d_1 + (1 - p_1)d_2 - (1 - p_1)c) = \delta_1$ which yields

$$p_1 = \frac{2(d_2 - c) - d_1 \pm \sqrt{(d_1 - 2(d_2 - c))^2 - 4(d_2 - c - d_1)(d_2 - c - \delta_1)}}{2(d_2 - c - d_1)}$$

provided $d_2 - c - d_1 \neq 0$. If $d_2 - c - d_1 = 0$ then $p_1 = \frac{\delta_1 - d_2 - c}{d_1 - 2(d_2 - c)}$.

Table 2.5: Example Prisoner's Dilemma Game

		Player 2	
		β_1	β_2
Player 1	α_1	d_1, d_1	$d_2, 0$
	α_2	$0, d_2$	c, c

Table 2.6: A game with Nash and limited-trust dominated strategies

		Player 2		
		β_1	β_2	β_3
Player 1	α_1	0,7	5,5	0,5
	α_2	3,2	5,4	7,1
	α_3	0,6	4,1	1,5
	α_4	2,1	3,10	1,0

2.3 Computation of 2-Player LTE(δ)

In this section we present a mathematical program for computation of an LTE(δ) in 2-player games. However, before doing so we consider the concept of a *dominated strategy* in a limited-trust game. By removing strongly dominated strategies, we will make the game smaller to aid in computation.

In a Nash game, a pure strategy s for player i is said to be dominated if $u_i(s, \sigma_{-i}) \leq u_i(s', \sigma_{-i})\ \forall \sigma_{-i}$ for some alternate feasible strategy s' which is a convex combination of player i's other pure strategies. s' is said to *weakly* dominate s if there is at least one σ_{-i} for which there is equality and at least on for which there is strict inequality. It is said to *strictly* dominate s if there is strict inequality for all σ_{-i}. In a limited-trust game with given δ, s' is said to dominate s if $u(s, \sigma_{-i}) \leq u(s', \sigma_{-i})\ \forall \sigma_{-i}$ and $u_i(s, \sigma_{-i}) \leq u_i(s', \sigma_{-i})$ for all σ_{-i}. Weak dominance occurs if there is some σ_{-i} for which $u(s, \sigma_{-i}) = u(s', \sigma_{-i})$. As with Nash equilibria, no LTE will have a player i playing a strictly dominated pure strategy s_i with nonzero probability. Also as in Nash equilibria, we can iteratively remove dominated strategies by examining each strategy individually to see if it is dominated by a convex combination of the other still present strategies (this is done by using a linear program).

Having introduced the idea of dominance in the limited-trust context, we now demonstrate it on the game in Table 2.6. From a Nash perspective, it is clear that α_2 strictly dominates α_3 and α_4 and weakly dominates α_1. Similarly, β_1 strictly dominates β_3. Therefore, these strategies need not be considered when looking for an MNE. From a limited-trust perspective this changes. α_2 no longer dominates α_1, α_3, or α_4 and β_1 no longer dominates β_3. Although no pure strategy dominates α_3, consider $\sigma_1 = [\frac{3}{5}, \frac{2}{5}, 0, 0]$. $u_1(\sigma_1, \beta_i) > u_1(\alpha_3, \beta_i)$ and $u(\sigma_1, \beta_i) > u(\alpha_3, \beta_i)$ for $i \in \{1, 2, 3\}$ so α_3 is still strictly dominated and can be dropped from the problem. However, α_4 is part of the socially optimal σ and therefore cannot be strictly dominated unless β_2 is strictly dominated first. While β_3 cannot be strictly or weakly dominated by a convex combination of β_1 and β_2 in the original problem, consider the problem after α_3 is removed. For a mixed strategy $\sigma_2 = [\frac{1}{5}, \frac{4}{5}, 0]$ we see that $u_2(\alpha_i, \sigma_2) > u_2(\alpha_i, \beta_3)$ and $u(\alpha_i, \sigma_2) > u(\alpha_i, \beta_3)$ for $i \in \{1, 2, 4\}$. Therefore, while we cannot remove β_3 immediately as in the Nash case, we can still remove it through the iterated removal of other dominated strategies. In the Nash case we then get the equivalent game in the left side of Table 2.7, and for the limited-trust case we get the equivalent game on the right side.

Table 2.7: Equivalent to the game in Table 2.6 for MNE's (left) and LTE(δ)'s (right)

		Player 2	
		β_1	β_2
Player 1	α_1	0,7	5,5
	α_2	3,2	5,4

		Player 2	
		β_1	β_2
Player 1	α_1	0,7	5,5
	α_2	3,2	5,4
	α_4	2,1	3,10

It is interesting to note that the value of δ is not relevant in determining whether a strategy is dominated in a game. This is because we cannot say that s' dominates s if $u(s, \sigma_{-i}) \leq u(s', \sigma_{-i})$ $\forall \sigma_{-i}$ and either $u_i(s, \sigma_{-i}) \leq u_i(s', \sigma_{-i})$ or $u_i(\sigma_i^G, \sigma_{-i}) - u_i(s', \sigma_{-i}) \leq \delta_i$ for all σ_{-i}. If the second condition occurs, s may still be part of a unique limited-trust best response. An example of this is given in Figure 2.4, in which, for a fixed σ_{-i}, player i has three strategies s_1, s_2, and s_3: despite the fact that $u(s_2, \sigma_{-i}) > u(s_1, \sigma_{-i})$ and $u_i(\sigma_i^G, \sigma_{-i}) - u_i(s_2, \sigma_{-i}) < \delta_i$, the limited-trust best response is a convex combination of s_1 and s_3, but not s_2.

Having defined limited-trust dominance to reduce computational effort, we now introduce our solution method. Our mathematical program for finding an LTE in a 2-player bimatrix game given by $A, B \in \mathcal{R}^{m \times n}$ will be loosely based on the linear program used in the support enumeration algorithm for

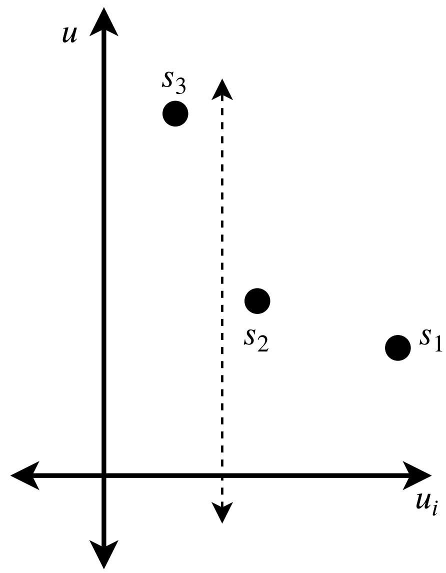

Figure 2.4: Image in which s_1 is part of limited-trust best response and s_2 is not

finding Nash equilibria, which determines if a given support pair S_A, S_B admits an MNE (i.e. there is an MNE in which only the pure strategies in S_A, S_B are played with positive probability, and all such strategies are played with positive probability). The mathematical program is given by MP1 where

$$f_x(y) = \max_x x^T(A+B)y \ \text{ subject to } \ x^T Ay \geq e_j^T Ay - \delta_1, \quad \forall j \in [m]$$

and $f_y(x)$ is similarly defined. MP1 constitutes a quadratically-constrained program with a bilevel component from $f_x(y)$ and $f_y(x)$. A, B are the $m \times n$ payoff matrices for players 1 and 2, respectively, and e_j is the vector with the value 1 at index j and zero elsewhere.

MP1:

$$
\begin{aligned}
&\max_{x,y} \quad 1 \\
&\text{subject to} \\
&\qquad x^T A y \geq e_j^T A y - \delta_1 && \forall j \in [m] && (1) \\
&\qquad x^T B y \geq x^T B e_j - \delta_2 && \forall j \in [n] && (2) \\
&\qquad x^T (A+B) y \geq f_x(y) && && (3) \\
&\qquad x^T (A+B) y \geq f_y(x) && && (4) \\
&\qquad \sum_{i \in S_A} x_i = 1 && && (5) \\
&\qquad \sum_{j \in S_B} y_j = 1 && && (6) \\
&\qquad x_i = 0 && \forall i \notin S_A && (7) \\
&\qquad y_j = 0 && \forall j \notin S_B && (8) \\
&\qquad x_i \geq 0 && \forall i \in [m] && (9) \\
&\qquad y_j \geq 0 && \forall j \in [n] && (10)
\end{aligned}
$$

The bilevel elements of constraints (3) and (4) are a necessary portion of the program: without these constraints, which force the total utility to be the greatest possible when each player is playing within δ_i of its greedy best response to the other, the solutions of MP1 would simply be a subset of the ε-equilibria for $\varepsilon = \max_i \delta_i$, regardless of whether they were also limited-trust equilibria. This cannot be solved using the objective function to drive the program, as the socially optimal ε-equilibrium is not necessarily an LTE. We also mentioned above that MP1 is loosely based on the Support Enumeration algorthim for finding 2-player MNEs. However, due to the fact that a general LTE(δ) is not a well-supported equilibrium, we are unable to fully linearize the constraints as in the support enumeration algorithm for finding MNE's. As a consequence, if $S_A \subseteq S_C$ and $S_B \subseteq S_D$, then any solution to MP1(S_A, S_B) is also a solution to MP1(S_C, S_D), which is not the case in the Nash support enumeration. The same problem is observed in finding non-well-supported approximate Nash equilibria as well, so this is not surprising.

We now prove below any LTE(δ) given by (x, y) is a solution to MP1 for appropriate S_A, S_B.

Theorem 6. *A strategy set (x, y) for a two player game is an LTE(δ) if and only if it is a feasible solution to MP1 for $S_A = [m]$, $S_B = [n]$.*

The proof of this Theorem may be found in A.1.2.

Corollary 1. *For any feasible solution to MP1, constraints (3) and (4) are fulfilled with equality.*

Proof. Follows from Theorem 6: any solution to MP1 is an LTE(δ), and any LTE(δ) fulfills the constraints with equality as each player is playing a limited-trust best response to the other. □

Given that MP1 was stated to have been loosely based on the linear program used in the support enumeration algorithm for Nash equilibria, it is natural to question why the program is not set up to iterate over supports, as in that algorithm. This comes about because in any greedy best response, every pure strategy which player i plays against the other player is a best response, and so the quadratic constraints (1) and (2) in MP1 can be transformed into a larger set of linear constraints which enforce the condition that every pure strategy in the support of a Nash equilibrium is a greedy best response. There is no corresponding condition for an LTE which allows us to consider the pure strategies of a support individually rather than the mixed strategy LTEs as a whole. However, if we are looking for well-supported LTE's we can use a support enumeration method be replacing constraints (1) and (2) in MP1 with those below and then apply Algorithm 1.

$$\begin{aligned}
e_i^T Ay &\geq e_j^T Ay - \delta_1 & \forall i \in S_A \quad \forall j \in [m] \\
x^T Be_i &\geq x^T Be_j - \delta_2 & \forall i \in S_B \quad \forall j \in [n] \\
x_i &= 0 & \forall i \notin S_A \\
y_j &= 0 & \forall j \notin S_B.
\end{aligned}$$

Algorithm 1 will find at least one LTE for every support pair S_A, S_B which admits a well-supported LTE. However, as we have already seen well-supported LTE's may not exist.

Algorithm 1 LTESupportEnumeration(A, B, δ_1, δ_2)

```
Initialize hashset LTESet ← ∅;
for S_A ∈ [m], S_B ∈ [n] do
    (x, y) ← SolveMP1(S_A, S_B);
    if (x, y) ∉ LTESet then
        LTESet[S_A, S_B] ← (x, y)
    end if
end for
return LTESet
```

2.4 Leader-Follower Equilibria

We have defined the concept of Limited-Trust equilibria in simultaneous games in a natural manner, and showed that at least one LTE exists in any simultaneous game of n players. The next natural extension to consider is LTE's in turn-based games, i.e. leader-follower or *Stackelberg* games.

Consider a two-player turn-based game of complete information, i.e. player 1 picks from m strategies and in response, player 2 picks from n strategies with full knowledge of the first player's choice. Such a game is akin to a bi-level optimization problem for the first player: given full-knowledge by all players, the second player's response is deterministically dictated by the first player. As such, this game always has a pure equilibrium known as the Stackelberg equilibrium and, assuming a fixed tie-breaking rule for players between multiple equivalent strategies, the Stackelberg equilibrium is unique. While this is true for n-player games, for the sake of simplicity we will confine our discussion to $n = 2$, as $n > 2$ follows naturally.

We now want to consider what happens when players have trust levels δ_1, δ_2. Although this is a full knowledge deterministic game with regard to the payoffs and the first player's action being known to the second player, unlike in the simultaneous game the nature of the equilibrium changes sharply depending on the first player's knowledge of δ_2 and the values from which each player measures δ_i. Because of this, we will examine three policies which represent different interpretations of a Limited-Trust Stackelberg Equilibrium (LTSE). We assume a bimatrix game with payoff matrices $A, B \in \mathcal{R}^{m\times n}$ for players 1 and 2, respectively. Note that here n is the number of pure strategies possessed by the second player.

1. Incomplete Knowledge: The first player does not know anything about

δ_2 and, being risk averse, assumes the second player is not trustworthy ($\delta_2 = 0$). The first player then determines the second player's response to each of its possible actions under this assumption and finds strategy i such that $i = \arg\max_{0 \leq j \leq m} a_{jr(j)}$ where $r(j)$ is player 2's best response to j and a_{ij} and b_{ij} are the first and second players' payoff if they play i and j, respectively. The first player then plays j which maximizes $a_{jr(j)} + b_{jr(j)}$ subject to $a_{ir(i)} - a_{jr(j)} \leq \delta_1$, and the second player plays l which maximizes $a_{jl} + b_{jl}$ subject to $b_{jr(j)} - b_{jl} \leq \delta_2$.

2. Complete Knowledge: The first player knows δ_2. It knows that if it plays i, then player 2 will play its best response $s(i)$ which maximizes $a_{is(i)} + b_{is(i)}$ subject to $b_{ir(i)} - b_{is(i)} \leq \delta_2$. Player 1 then finds i such that $i = \arg\max_{0 \leq j \leq m_1} a_{js(j)}$, and plays j which maximizes $a_{js(j)} + b_{js(j)}$ subject to $a_{is(i)} - a_{js(j)} \leq \delta_1$. The second player then plays $s(j)$.

3. Cooperative Complete Knowledge: Let i, j be the regular Stackelberg Equilibrium. The players play k, l which maximizes $a_{kl} + b_{kl}$ subject to $a_{kl} - a_{ij} \leq \delta_1$ and $b_{kl} - b_{ij} \leq \delta_2$.

Of these policies, the first two seem like the most natural interpretations of the LTE in the turn-based game: a player is willing to forgo a payoff at most δ_i higher than what they could get, provided that the other player gains at least that much. The only question is whether or not player 1 knows δ_2: while the question was unimportant in the simultaneous setting as equilibrium was merely a point where no player could unilaterally improve the total utility without exceeding its maximum cost, here the leader-follower nature of the game means the first player can determine exactly how the second player will act and plan its strategy accordingly. The only question for the first player is the value of δ_2; if it is unsure then it plans for the worst and assumes $\delta_2 = 0$.

While the cooperative complete knowledge policy may seem less natural, the confusion is a matter of perspective: with the first player having full knowledge of δ_2, instead of measuring its payoffs over the second player's reactions to each strategy i it could play, it instead measures them with respect to the Stackelberg equilibrium. The second player makes the same choice: it is rational and can determine the first player could have played according to the Stackelberg equilibrium if it wished to, and so reacts accordingly. This policy requires more coordination between players, but can be interpreted as two players who regularly interact and strive to maintain a good relationship. In this sense it is less suited for one-off games. However, the same could be said of the complete knowledge policy, as it is otherwise infeasible to expect the first player to know δ_2 *a priori*.

We have derived additional results in regard to the expectations and probabilities for all three of these policies in random leader-follower games. However, the details are somewhat involved and do not provide any great insight to the reader. As such, these results and their derivations are available in A.2.

2.5 Numerical Results

In this section we present a numerical comparison of the efficiencies of the LTE when compared to Nash equilbria in both the simultaneous and leader-follower settings. LTE's are found for randomly-generated games and compared with the maximum-value Nash and Stackelberg PoA's of these games. These represent random repeated games, a set of games which, while not identical, are all drawn from the same distribution. These games are of particular interest because while the LTE is explicitly created for analyzing one-off games, it is implicitly motivated by the expectation that future games will be played. Day-to-day societal interactions are a perfect example of this, and are well modeled by random repeated games: such interactions between players are not identical, but will display a pattern over time so that they could be said to come from some "typical" distribution.

Theoretical results related to random repeated games in the leader-follower setting can be viewed in detail in A.2.3, but one which we will state here is that the expected PoA of the Stackelberg equilibrium for a 2-player leader-follower bimatrix game. In such a game with $m \times n$ payoff matrices where each entry generated is generated independently and from and identical distribution (iid) A or B for players 1 and 2, respectively, the expected PoA is

$$\frac{E[(A+B)^{(mn)}]}{E[A^{(m)}]+E[B^{(n)}]}$$

where $X^{(n)}$ is the maximum of n samples x_i generated iid from distribution X.

We consider random repeated games in our numerical trials. These are represented as bimatrix games where the payoff matrices A, B for each player are generated according to some distribution. In particular, we consider matrices where each entry is generated iid for each player, though A and B may not come from the same distribution. The majority of games were generated as 2-player 2×2 repeated games, with entries of player 1's payoff matrix generated iid according to a distribution A and player 2's payoff matrix generated iid according to a distribution B. One instance of 2-player 3×3 games was generated, as unlike in the leader-follower games we have a less precise bound on the PoA as a function of m and n. LTE's of simultaneous games were computed using MP1 in Section 2.3, and LTSE's were computed for each leader-follower policy using the methods given in their descriptions in Section 2.4.

2.5.1 Simultaneous Games

We consider games where players' payoffs are generated iid from three distributions: geometric with $p = \frac{1}{4}$, uniform over the set of integers in $[0, 10]$, and Normal $\mathcal{N}(0, 1)$. 100 instances of 2×2 games are generated from each of these distributions, as well as 100 3×3 games drawn from the geometric distribution. We then vary $\delta_1 = \delta_2 = \delta$ from 0.01 to 1 for each game, to see how the value of the LTE changes as a function of δ. For each test case, we use the support enumeration method to determine the MNE with the highest net utility for comparison to the LTE with the highest net utility. Because any MNE which provides the optimal level of social utility is also an LTE, we ignored generated test cases where the social optimum was also an MNE.

Because of the nonconvex and potentially disjoint nature of the solution space, we include a constraint in MP1 that the value of the net utility of the LTE must be greater than or equal to the utility provided by the best MNE (net utility-maximizing MNE). As we have already seen in Section 2.2, such an LTE may not exist for all values of δ. Therefore, if the solver fails to locate a feasible solution to MP1 after 50 attempts on a particular test case, this constraint is relaxed. It is not reintroduced until after δ has increased to a level where an LTE with better net utility than the MNE is discovered. Additionally, while we compute an LTE which has higher net utility than all

MNE's, it may not necessarily be the maximum value LTE. This is due to the nonconvexity of the set of LTE's.

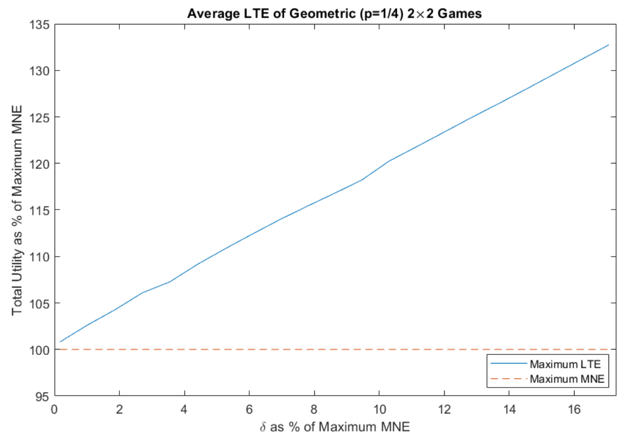

Figure 2.5: 2×2 Geometric Games

Figures 2.5 through 2.8 detail the results of our numerical trials. Although δ is always varied from 0.01 to 1, in the figures it is rewritten as the percentage of the net utility generated by socially optimal (net utility-maximizing) MNE so as to compare values across different distributions. In all but Figure 2.7, we see a very clear linear relationship between δ as a percentage of the maximum-valued MNE and LTE as a percentage of the MNE. In both the geometric games, the curve has a slope of approximately 2, meaning that on average, for every game a player has to give up δ, there is a game where it gains 3δ over what it would receive by playing selfishly. The uniform games in Figure 2.8 show a similar result, with a slope of approximately 1.3.

This brings us to Figure 2.7, which unlike the others does not evince an approximately linear curve. However, consider the variance of the distributions: the geometric distribution with $p = \frac{1}{4}$ has a variance of 12 and the discrete uniform distribution over $[0, 10]$ has a variance of 10. In contrast, the variance of 1 possessed by the Normal distribution is quite small. Now consider what the curves in Figures 2.5, 2.6, and 2.8 would look like if we continued to increase δ: the curves would eventually start to evince diminishing returns, as increasing δ past the point where many social optima start to become LTE's will produce very little additional social utility. This explains the curve in Figure 2.7: it is merely a curve in which the δ is already quite large compared to the variance of the distributions from which the entries of A and B are generated, and thus is experiencing diminishing returns. It

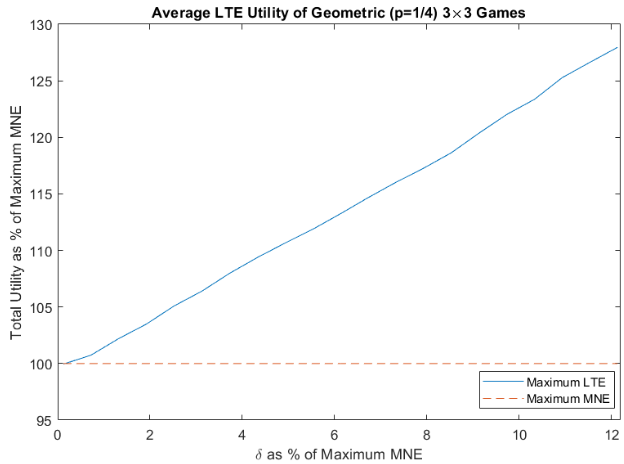

Figure 2.6: 3×3 Geometric Games

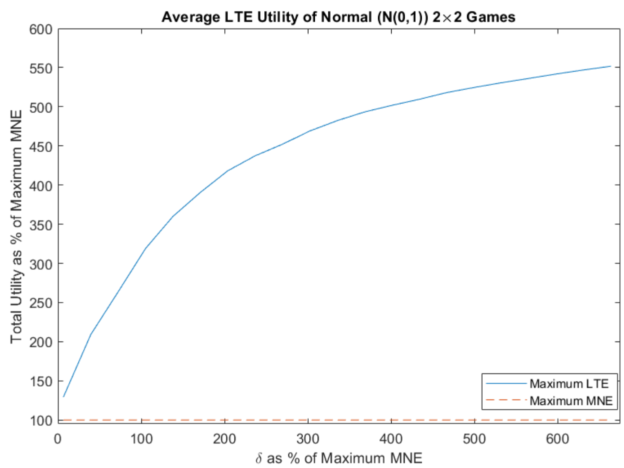

Figure 2.7: 2×2 Normal Games

also indicates that if we continue to increase δ, the other figures will come to resemble Figure 2.7.

2.5.2 Leader-Follower Games

We conduct numerical studies on 2×2 games with payoff matrices generated from three distributions: $U[-0.5, 0.5]$, $\mathcal{N}(0,1)$, and $exp(1)$. For each set of trials, we let $A \sim B$, and let $\delta_1 = \delta_2 = \delta$. We define the Stackelberg gap as the difference between the Stackelberg equilibrium and the social optimum. Figures 2.9, 2.10, and 2.11 each show the average PoA of 1000 games generated according to these distributions and solved for $\delta \in [0,1]$

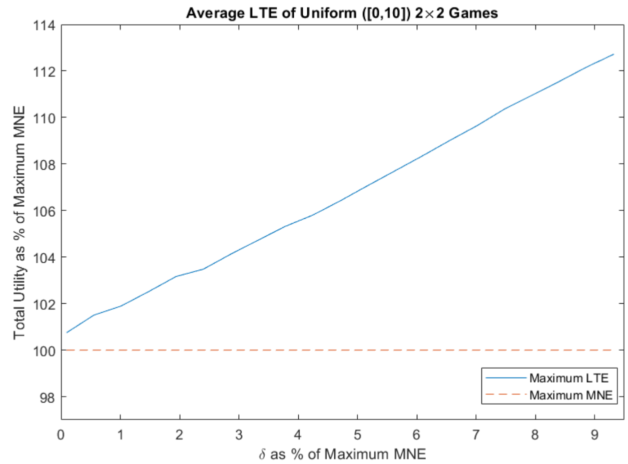

Figure 2.8: 2×2 Uniform Games

in the first graphs, where as mentioned in Section 2.4, $\delta = 0$ indicates that there is no trust between the players. The second graph considers how much of the Stackelberg gap is covered by each of the policies at the varying δ levels. Unlike in the simultaneous case, in the leader-follower setting under the complete knowledge policy if the Stackelberg equilibrium is the social optimum that does not guarantee it is also the LTSE. For that reason we have not ignored games in which this occurs.

Unsurprisingly, in all three distributions for all values of δ, the cooperative complete knowledge policy results in the best performance. We even note that with $\delta = 0$ it still manages to recover an average of approximately 20% or greater of the Stackelberg gap for each tested set of games. This "Reward without Risk" comes from the greater cooperation between players seen in this interpretation of the LTE.

Also unsurprisingly, the complete knowledge policy tends to outperform the incomplete knowledge policy on average, for most δ values. Figure 2.9 provides an excellent demonstration of this: because the max and min possible payoffs have a gap of 1, by the time $\delta = 1$ both players are trying only to maximize the social utility. In particular, by the time δ reaches approximately 0.75, the complete knowledge game tends to result in the social optimum being played virtually every time. This is because the entries of A and B are drawn from $U[-0.5, 0.5]$ so the chance of the socially optimal outcome having a utility for player 1 which is more than 0.75 less than the player's greediest move is nearly 0. In contrast, while the same is true in the

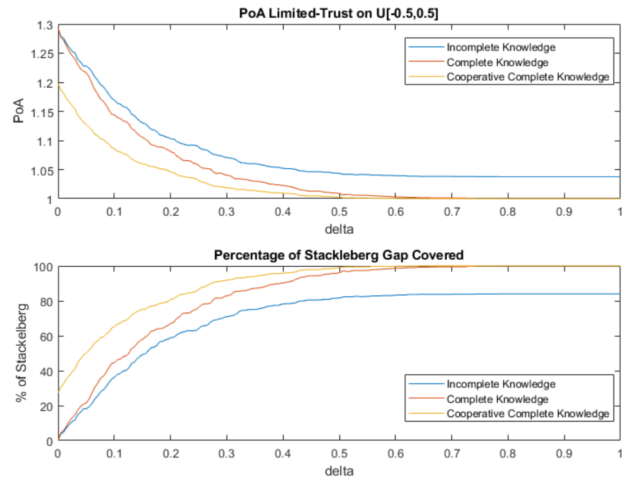

Figure 2.9: Leader-Follower Numerical Results, $A \sim B \sim U[-0.5, 0.5]$

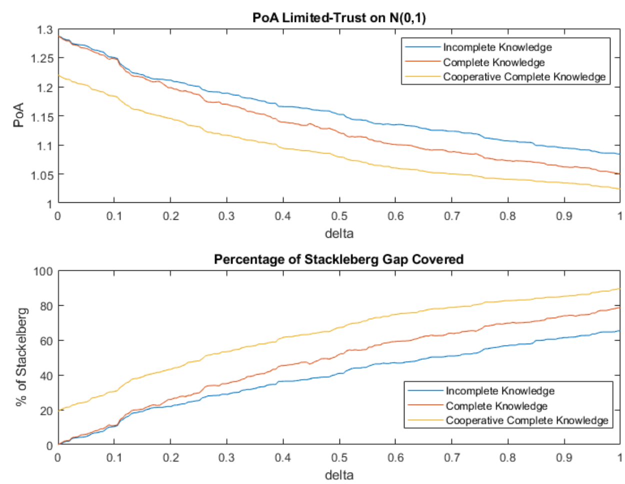

Figure 2.10: Leader-Follower Numerical Results, $A \sim B \sim \mathcal{N}(0, 1)$

incomplete knowledge case, the game levels off to covering slightly over 80% of the Stackelberg gap even at $\delta = 1$. This occurs due to the fact that although both players are effectively altruistic at this level of δ, the first player does not believe that the second player is. This causes player 1 to attempt to maximize the social utility around the assumption that $\delta_2 = 0$, despite the fact this is not true. We can consider this gap between the incomplete and complete knowledge cases as the cost of ignorance.

It is important to note that the cost of ignorance may not be bad. Indeed Figure 2.11 shows that for δ between approximately 0.05 and 0.3, the cost of ignorance is negative. This occurs due to the fact that the first player is

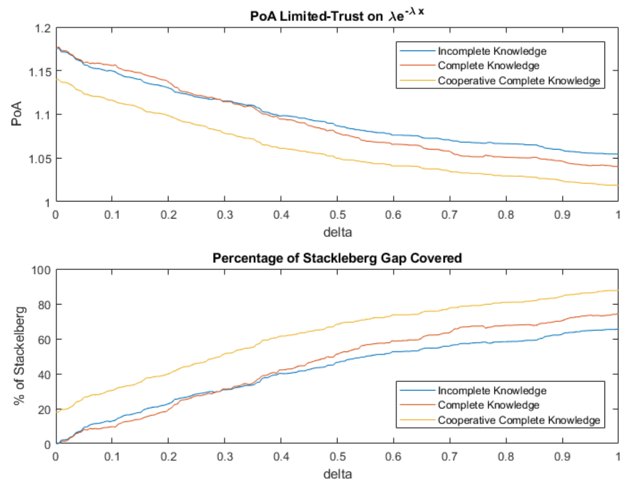

Figure 2.11: Leader-Follower Numerical Results, $A \sim B \sim e^{-x}$

unaware that the second player is willing to give up δ, and thus is unable to take advantage of that fact for its own gain. This is identical to what happens for some values of δ in the game described in Table A.1 in A.2.1.

2.6 Discussion and Summary

Throughout this paper we have been considering limited-trust equilibria as a description of behavior which is not entirely selfish, provided the opportunity cost of the behavior for player i is less than some bound δ_i. This idea of an LTE was expressed very naturally in simultaneous games, where at equilibrium each player does not care about the δ values which are motivating other players, only that it plays its best response to what those players are actually doing. The key managerial insight of the LTE is that while the players in giving something up appear to be playing "non-rationally" when games are considered in isolation, when considered as a whole both players actually achieve more than they would have received if they had myopically played the "rational" Nash equilibria in each game. We saw that while it was possible for limited-trust games to have worse results than Nash games, it will not happen in several common classes of games, and occurs rarely in others: in 2-player numerical trials with $\delta_1 = \delta_2 = \delta$ we observed an average personal utility increase of δ for each player when δ was modest compared to the variance in the utilities of randomly generated games. When we consider the leader-follower setting players can no longer ignore the δ values of their fellows, and we considered the effects of whether or not players knew each other's δ's or had to prepare for the worst (assume $\delta_{-i} = 0$).

It is natural to question the method developed in this paper for the computation of LTEs in a simultaneous game. Given that LTEs are a subset of ε-equilibria, which are PPAD-hard to compute for general ε, we do not expect to derive an algorithm for the general k-player case without a mathematical program similar to MP1. However, readers may wonder why we have not provided a different algorithm for computing an LTE in the 2-player game.

The Lemke-Howson algorithm [34] is one of the first algorithms for finding Nash equilibria in a 2-player bimatrix game and remains one of the most popular. It relies on the observation that at equilibrium (σ_1^G, σ_2^G), if player i has m_i pure strategies then for a best response $\sigma_i^G = \{p_1^i, p_2^i, ..., p_{m_i}^i\}$ either $p_j^i = 0$ or playing the pure strategy s_j^i is a best response to σ_{-i}. With this observation, the Lemke-Howson algorithm is able to set up a linear complementarity program (LCP) for which any feasible solution is a Nash equilibrium. Unfortunately the definition of an LTE does not lend itself well to this method. This is partially due to the fact that we cannot make a similar observation about pure strategies in an LTE. However, while this problem may be possible to overcome, the larger difficulty comes from the fact that there is an optimization problem embedded in each player's best response to the other. While this is also true of a greedy best response, that optimization problem can be expressed solely as a set of linear constraints with no objective function, i.e. σ_i is a best response to σ_{-i} if and only if $u_i(\sigma_i, \sigma_{-i}) \geq u_i(e_j, \sigma_{-i})$ for all $j \in [m_i]$. The optimization problem embedded in the limited-trust best-response cannot be absorbed to a larger program due to having an objective function. This is reflected in the fact that the best response function is explicitly brought into MP1 in constraints (3) and (4), rather than bringing in constraint sets. Even the further generalization of the algorithm in [35] is unlikely to adapt to computing LTEs. Although the Lemke-Howson algorithm is nearly sixty years old and has since been shown to be a special case of the Global Newton Method by [36], it remains an extremely prevalent method for computing Nash equilibria in 2-player finite games in practice. This is particularly true following the proof by [37] that ε-equilibria (and Nash equilibria) are PPAD-complete to compute even for 2-player finite games. It is worth noting that as a consequence, LTE(δ) is also PPAD-hard to compute.

We also considered several natural interpretations of the LTE in a leader-follower game, which vary drastically depending on how much knowledge players have of each other. More definite theoretical probabilities for the likelihood of a social optimum occurring in a random game in the leader-follower context, as these equilibria are significantly easier to compute. We then moved onto numerical testing of the LTE, comparing how the social utility varied over random repeated games as a function of δ, particularly when compared to Nash and Stackelberg equilibria. One of the more surprising results of our numerical trials in simultaneous games was how strong the linear relationship was between the net utility and δ, prior to the onset of diminishing returns as δ continues to increase. In our leader-follower games we observed the differences in the utility of each of our interpretations, noting that the cooperative complete knowledge case produced significant gains at the no risk level of $\delta = 0$, and also that the gap between complete and incomplete knowledge effectively measured the price of ignorance. Perhaps more surprising was that the price of ignorance was sometimes negative on average, rather than just occasionally, with parameters existing for which player 1 assuming the worst of player 2 resulted in higher average total utility.

As noted earlier in this section, while many traditional equilibrium computation methods such as the Lemke-Howson algorithm are unlikely to adapt well to the LTE computation, we would still like to put more study into the computation of simultaneous game LTE's. Additionally, we are interested in considering how LTE's model behavior in larger systems such as social networks. Perhaps the most exciting line of inquiry is that of learning: the LTE is positioned as a tool for non-Nash analysis of repeated game that can also solve one-off simultaneous games, something for which there are few existing tools. As such each player should be trying to set their δ_i in order to maximize their utility over time. We are very interested in the potential dynamics of shifts in δ values as players interact with each other, particularly if they take each other's playing history into account. Also of interest is the relationship between talented or well-connected but relatively selfish individuals as opposed to trustworthy individuals without specialized skills, and the resulting "diva" behavior often exhibited by the former. We will focus on these areas of study in Chapter 3.

CHAPTER 3

TRUST IN SOCIAL NETWORK GAMES: THE BENEFITS OF RECIPROCITY

A paper based on the work in this chapter has been submitted to *IEEE Transactions on Network Science and Engineering*.

3.1 Introduction

Choose your friends wisely. It's good advice and it also applies to the problem of selecting partners to work with. Developing corporate partnerships and alliances is a long-standing business practice, with many modern successful examples such as GoPro and Redbull, Amazon Inc and the US Postal Service, and Disney and Pixar (pre-acquisition), just to name a few. While strategies for corporate partnerships are well-studied, the problem of forming interpersonal partnerships is more nebulous. However, a vital factor in partnerships between peers is trust. Suppose you have put together a project which will require collaboration from a colleague, and you must decide between two potential collaborators. Both candidates possess the same basic level of expertise, leading you to expect that the project will be a success with either of them, but one candidate has a reputation for taking all of the credit in collaborations and using them to advance their own interests over those of their partner. As the head of the project, you would prefer to avoid working with this individual, and instead collaborate with your other colleague who has a reputation for being upfront and fair during collaborations.

In this paper we consider agents in a social network who have the opportunity to interact and benefit from each other. These interactions occur as limited-trust leader-follower games, where limited-trust (and associated equilibria) is a concept recently developed by [3]. Loosely speaking, the limited-trust concept assumes that an agent will help its fellow agent, provided that the cost is not too high and is outweighed by the utility provided to the other players. The trust level of an agent i is determined by a metric $\delta_i \geq 0$ so that if $\delta_i = 1$, agent i is willing to give up \$1 provided it helps its partner agent j receive more than \$1 extra. The concept is natural, and players in 2-player limited-trust games both benefit in the long run when playing in this way. However, it is not standalone in the same way that concepts like grim trigger strategies and the Nash equilibrium are: in any given interaction, a player can only increase its utility by being more selfish if all other players have fixed δ. When limited-trust is viewed in the context of agents who must compete to attract partners though, we will see that self-interested agents engage in strongly trustworthy behavior, as doing otherwise causes them to miss out on partnership opportunities.

The main contribution of this paper is the development of a system for modeling interactions between individuals in a social network. This system is thoroughly analyzed, with algorithms developed for individuals to learn how trustworthy their neighbors are and how to alter their own trust level accordingly. As we will see later, these behavioral changes increase trustworthiness which leads to a substantial increase in the total utility across the entire network. Empirically, such a system is substantiated by numerous studies in evolutionary biology and psychology whose results mirror the behaviors which are naturally occurring in the model. These results indicate that the most successful individuals are those who are willing to behave in a trustworthy manner. What they lose in individual interactions they more than make up for by increasing their opportunities. Further, myopic agents naturally arrive at trustworthy behavior without the use of external history-based mechanisms such as grim trigger or tit-for-tat strategies. As such, we feel that the flexibility of limited-trust is more natural and intuitive for the social scenario we consider.

Our empirical studies also reveal a counter-intuitive managerial insight. While it would be natural to expect extreme competitiveness and selfishness when opportunities are limited and individuals try to make the most of them, we actually observe the opposite and find that individuals are at their most trustworthy when opportunities are limited. Indeed, individuals do their best to appear trustworthy in order to capture the few available opportunities. Similarly, it is also natural to expect individuals to be more trustworthy when there are more opportunities, as taking advantage of any one opportunity is not worth the resulting reputational damage. Instead, it is surprising to observe that the relative glut of opportunities outweighed the reputational consequences for selfish individuals, as they knew that their behavior would not limit their opportunities (subject to network structure).

We next consider a motivating example, given by the social network in Figure 3.1. Suppose that each agent has to lead $k = 2$ leader-follower games and can select any 2 neighbors as followers for these games, which are randomly drawn from a known distribution. Let the games be over 2×2 payoff matrices with each entry drawn independently and identically at random from an exponential distribution where $\lambda = 2$. If all agents play selfishly, with $\delta_i = 0$ for every agent i, then each agent will select its partners uniformly at random. That means that agent 7 can expect to be chosen as a follower for $\frac{17}{6}$ games, in addition to the 2 games where it is a leader. Suppose that agent 7 now behaves in a slightly trustworthy manner, with $\delta_7 = 0.01$ whenever it interacts with any other agent, while all of its neighbors remain selfish: it can now expect to attract games from all of its neighbors, and participate in 4 games as a follower. Doing so it will likely achieve less utility per game, but it will play in an additional $\frac{7}{6}$ games over what it would otherwise expect. Table 3.1 compares these two settings, showing that when all other agents are selfish, agent 7 can improve its average utility by over 25% by being only slightly trustworthy! The table also reveals a second managerial insight: when all agents are equally trustworthy ($\delta_i = 2$ for $i \in \{1, 2, ..., 7\}$), all stand to gain a significant amount of utility (approximately 14.5% per agent) and hence drive a significant increase in the net utility of the system.

After defining the model in Section 3.2, we will return to this example to understand how network position interacts with the distribution of δ_i across players. We will see that generally speaking, having a comparatively high value of δ leads to an agent increasing its utility by increasing its number of interactions. We also note that while in this paper we will constrain agent i to express a single value of δ_i to each of its neighbors for the sake of simplicity, in Section 3.4.1 we will cover how agent i can optimally set personal values $\delta_i(j)$ for each of its neighbors j.

This paper is organized as follows: the rest of this section conducts a literature review over relevant work to our topic, particularly on the subjects of network games and evolutionary biology and psychology. In Section 3.2 we formally define the mechanics of our model and how agents within it behave in a full knowledge setting while in Section 3.3 we define the same functions

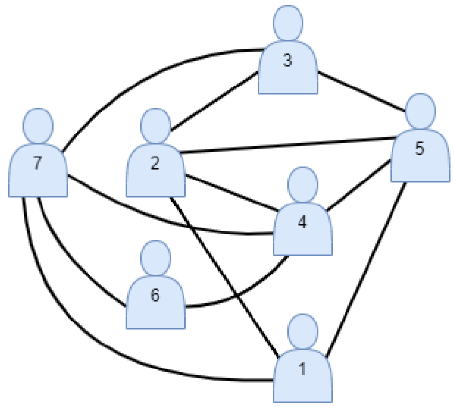

Figure 3.1: Example Social Network

Table 3.1: Comparison of utilities for different values of δ for the network in Figure 3.1

		$\delta_i = 0$	$\delta_i = 0 \quad \forall i \neq 7$ $\delta_7 = 0.01$	$\delta_i = 2$
Player	Degree	Utility per Round	Utility per Round	Utility per Round
1	3	10.532	10.743	11.974
2	4	12.880	11.469	14.935
3	3	10.596	10.356	12.271
4	4	13.420	13.395	15.479
5	4	13.100	11.499	14.716
6	2	9.178	8.570	10.289
7	4	14.338	18.025	16.531
Average		12.012	12.014	13.749

for agents with incomplete information and knowledge of their fellows. In Section 3.4 we consider the metagame which occurs on top of the model when agents are able to adjust their levels of trustworthiness, before exploring the model numerically in Section 3.5. In Section 3.6 we discuss the numerical results and future directions of our work, before concluding our paper in Section 3.7. Additionally, we provide multiple appendices for exploration of additional topics related to our system, and proofs of some theorems.

We note that readers who are primarily interested in the results of our paper may wish to skip Sections 3.3 and 3.4: while these sections are necessary to explain how the system operates when players have incomplete information or vary their level of trustworthiness, they are in-depth descriptions of functions which can be grasped intuitively.

3.1.1 Literature Review

Explaining and modeling unprompted generosity has been an intriguing question within Game Theory, one apparently at odds the widespread idea of Nash equilibria [1]. [29] provides one explanation in the form of α-altruism. α-altruism is inspired by Hamilton's rule for kin selection ([4]) which loosely states that we help others because they are some portion of ourselves on a genetic level. α-altruism incorporates this by computing perceived costs for each player, which are a convex combination of the player's personal cost and the net cost for all players. Hamilton's rule applies only to kin though, both theoretically and empirically. Thus α-altruism is less solidly grounded outside of this setting.

However, evolutionary biology offers another explanation for unprompted generosity: partner selection. Studies such as [38–43] consider various settings under which participants engage in 2-stage interactions: while partnered randomly during the first stage, participants may decide who to partner with in the second stage. In each study, participants who were more generous in the first stage were more desirable as partners in the second, and were more likely to be generous in the first stage if they knew beforehand about the second stage. [39] also finds that generosity may be faked in the first round to take advantage of the partner in the second round. [44] conducts an empirical study which demonstrates that generosity and cooperation only tend to arise between partners of relatively similar opportunities (for selection of partners). [45] empirically tests partner selection as a motivation for generosity with a competing theory, threat premium, which states that individuals are generous in order to avoid potential conflict or danger. The study finds that partner selection is a far stronger motivator.

This paper makes use of the LTE from Chapter 2. We noted then that the LTE is explicitly motivated by partner selection and in this paper it is applied to study how partnerships form within social networks. Social networks are a frequent topic of study in evolutionary game theory, with papers such as [46–51] studying how coalitions and cooperative behaviors can form naturally within social networks under various settings and assumptions. [52] is particularly interesting under this evolutionary setting, as it studies the problem of seed selection for cooperative behavior, i.e., how many people must play cooperatively in order to trigger a cascade of cooperative behavior throughout the network. On a related topic, [53] provides a large scale experiment which successfully identifies network structures which increase the effect of peer influence. For the interested reader, [54] provides a good primer on evolutionary game theory through 2007, and [55] provides an extensive survey over a much larger class of games in social networks through 2015.

However, none of the evolutionary game theory papers mentioned above study partner selection in conjunction with social networks. To the best of our knowledge the only paper which does so is [56]. It finds that frequent partner switching helps to dissuade defection in the prisoner's dilemma, as individuals who take advantage quickly find that no one is willing to play with them. As in our setting, players make partner selections based on reputation and past observations of their two-hop neighborhoods. However, these agents select their partner groups by updating their one- and two-hop neighborhoods within the network, and update their reputations by mimicking their successful neighbors rather than playing best responses to their neighbors. Additionally, in this chapter we consider a broader class of games generated from arbitrary distributions. These can model any interactions necessary including the prisoner's dilemma considered in [56]. We consider these games within the limited-trust setting, and find our results congruent with the recent papers on partner selection mentioned above.

3.2 Game Model

3.2.1 Preliminary Concepts

Before defining the systems we consider, we begin with a review of some standard concepts in game theory.

Definition 6 (Strategy Profile of a Finite Game). *Given a finite N-player game in which each player i has a set Σ_i of pure strategies, a valid pure strategy profile for the game is given by $\sigma = \{\sigma_1, \sigma_2, ..., \sigma_N\}$ where $\sigma_i \in \Sigma_i$ is the pure strategy played by agent i.*

Definition 7 (Stackelberg Equilibrium). *A 2-player leader-follower (Stackelberg) game is said to display a pure Stackleberg equilibrium $\{\sigma_1, \sigma_2\}$ when player 2 is playing its best response to player 1's strategy, and any deviation* *by player 1 from σ_1 to a new strategy σ'_1 will not increase its utility after player 2 makes its best response σ'_2 to σ'_1.*

While it is common to consider mixed strategies, we will be considering leader-follower games with full knowledge. All such games posses pure-strategy Stackelberg equilibria, and any mixed strategy equilibria are convex combinations of pure equilibria. Therefore, we will not need to consider mixed strategies in this paper. Additionally, although the Stackelberg equilibrium can be easily extended to the N-player setting, we will consider only one-on-one interactions between players which eliminates the need for this extension.

We are interested in a related concept to the Stackelberg equilibrium, the Limited-Trust Stackelberg Equilibrium (LTSE) ([3]). The LTSE similarly possesses pure-strategy equilbria, and will govern player interactions. A limited-trust game is a game in which each player i has a trust-level $\delta_i \geq 0$ which it is willing to give up from its greedy best-response (the strategy which maximizes its own utility given the strategies of all other players) provided that doing so increases the net utility of all players. For example, consider the 2-player game in Table 3.2 in which player 2 must decide between two strategies a_2 and b_2. Suppose that player 1 has selected a_1, and so player 2 must decide between $u_1(a_1, a_2) = 4, u_2(a_1, a_2) = 3$ if it selects a_2 and $u_1(a_1, b_2) = 2, u_2(a_1, b_2) = 4$ if it selects b_2. Suppose that $\delta_2 = 2$. Then the

Table 3.2: Example 2×2 game

		Player 2	
		a_2	b_2
Player 1	a_1	4,3	2,4
	b_1	3,2	1,3

second player's best response to the first player is to play a_2, as it maximizes net utility $(4 + 3 > 2 + 4)$ and results in a loss of 1 from player 2's greedy best response, which is acceptable given $\delta_2 \geq 1$. This allows all players to benefit in the long run by avoiding inefficient equilibria which only benefit one player. In a 2-player leader-follower game the limited-trust best response of the follower to the leader playing $s_1 \in \Sigma_1$ is

$$\begin{aligned} r_2(s_1, \delta_2) = \arg\max_{s_2 \in \Sigma_2} \quad & u_1(s_1, s_2) + u_2(s_1, s_2) \\ \text{s.t.} \quad & u_2(s_1, G_2(s_1)) - u_2(s_1, s_2) \leq \delta_2, \end{aligned}$$

where $G_2(s_1) = \arg\max_{s_2 \in \Sigma_2} u_2(s_1, s_2)$ is the follower's greedy best response. $r_2(s_1, \delta_2)$ is therefore the strategy which maximizes net utility, subject to the constraint that the follower does not give up more than δ_2 than it could have obtained from the greedy best response. The leader's limited-trust optimal strategy is

$$\begin{aligned} s_1^*(\delta_1, \delta_2) = \arg\max_{s_1 \in \Sigma_1} \quad & u_1(s_1, r_2(s_1, \delta_2)) + u_2(s_1, r_2(s_1, \delta_2)) \\ \text{s.t.} \quad & u_1(G_1(\delta_2), r_2(G_1(\delta_2), \delta_2)) - u_1(s_1, r_2(s_1, \delta_2)) \leq \delta_1, \end{aligned}$$

where $G_1(\delta_2) = \arg\max_{s_1 \in \Sigma_1} u_1(s_1, r_2(s_1, \delta_2))$ is the leader's greedy best strategy, given the limited-trust best response which will be made by the follower.

Next, we define the 2-player LTSE, which will be relevant for this paper, and the N-player extension can be easily intuited from this and the definition of a Stackelberg equilibrium.

Definition 8 (Limited-Trust Stackelberg Equilibrium)**.** *A strategy pair (s_1, s_2) in a 2-player limited-trust Stackelberg game with trust levels δ_1, δ_2 is said to be a limited-trust Stackelberg equilibrium if and only if $s_1 \in s_1^*(\delta_1, \delta_2)$ and $s_2 \in r_2(s_1, \delta_2)$ (if each player plays its Stackelberg limited-trust best response or strategy to the other).*

Note that when $\delta_1 = \delta_2 = 0$, the LTSE reduces to a Stackelberg equilibrium.

3.2.2 System Model

Having covered the preliminaries, we now introduce our model for interactions in a social network. We consider a system over a social network $G(V, E)$ with a set of vertices V and edges E. Vertices represent agents in the network and an edge between vertices means that the corresponding agents can interact. Each agent in G is self-interested and seeks to maximize its own utility. However, direct interactions between agents occurs only in a one-on-one setting through 2-player limited-trust Stackelberg games. As such, each agent i has a trust level δ_i which governs its individual interactions with other agents. The system as a whole can be considered as an N-player utility maximization game, where $N = |V|$.

Let N_i^1 represent the one-hop neighborhood of agent i in G: N_i^1 is the set of all agents j for which $(i, j) \in E$. Let

$$N_i^2 = \left(\bigcup_{j \in N_i^1} N_j^1 \right) \setminus (N_i^1 \cup \{i\})$$

represent the 2-hop neighborhood of i, the set of agents who are not in N_i^1 but have neighbors in N_i^1. For one time period in the system, agent i may invite at most k_i of its fellow agents in N_i^1 to interact, where $k_i \in \mathcal{Z}^+$. If agent $j \in N_i^1$ accepts an invitation from i, they engage in a leader-follower game with leader i and follower j over payoff matrices A and B, respectively, such that $A \sim \mathcal{A}_{ij}$ and $B \sim \mathcal{B}_{ij}$ where $\mathcal{A}_{ij}, \mathcal{B}_{ij}$ are probability distributions for interactions between i and j initiated by i.

Although agent i may issue at most k_i invitations per time period, it is allowed to accept as many as it receives. This consideration is motivated by the fact that it is easy for an individual to take a supporting role in many endeavors, but it generally only has the time or resources to take a lead role on a small number. Further, while i may both accept an invite and receive an invite from a neighbor j, leading to two separate interactions, it may not issue multiple invitations to j within a single time period. This means that agent i may have up to a maximum of $k_i + |N_i^1|$ interactions per time period. An interaction in which i invites j can thus be fully characterized by $\theta_{ij} = \{\mathcal{A}_{ij}, \mathcal{B}_{ij}, \delta_i, \delta_j\}$, with expected utilities $u_i(\theta_{ij}), u_j(\theta_{ij})$ for each player. Agent j will accept i's invitation provided $u_j(\theta_{ij}) \geq 0$. i will choose to invite (at most) k_i of its neighbors, with a neighbor j being selected if it provides one of the k_i highest values for $u_i(\theta_{ij})$ in N_i^1. This is subject to $u_i(\theta_{ij}), u_j(\theta_{ij}) \geq 0$ as otherwise either it does not benefit i to interact or it does not benefit j to accept the invitation.

Given the methods that i uses to choose which neighbors to invite and which invitations to accept, we can characterize all behavior in the system if we know $\theta = \{G, \mathcal{A}, \mathcal{B}, \delta\}$ where $\mathcal{A} = \{\mathcal{A}_{ij}\}_{(i,j)\in E}$, $\mathcal{B} = \{\mathcal{B}_{ij}\}_{(i,j)\in E}$, $\delta = \{\delta_i\}_{i\in[N]}$. Let K_i^1 be the set of neighbors that i invites to interact and K_i^2 be the set of neighbors that invite i to interact in a time period. Agent i's expected total utility is

$$\mathbf{u}_i(\theta) = v_i(\theta) + w_i(\theta) \enspace ,$$

where $v_i(\theta) = \sum_{j\in K_i^1} u_i(\theta_{ij})$, the value of the games i initiates which are accepted, and $w_i(\theta) = \sum_{j\in K_i^2} u_i(\theta_{ji})$, the value of the games i accepts invitations to. Note that K_i^1 can be determined entirely from knowledge of N_i^1, and K_i^2 can be determined from knowledge of $N_i^1 \cup N_i^2$, meaning that agent i's interactions depend only on its 1- and 2-hop neighborhoods, not the network as a whole.

Lemma 2. *Given a 2-player limited-trust Stackelberg game between a leader i and a follower j, $u_j(\theta_{ij})$ increases monotonically as δ_i increases.*

Proof. Consider a follower j with fixed δ_j. For any action s_i the leader i takes, j has a deterministic response $r_2(s_i, \delta_j)$. Note that r_2 is not a function of δ_i, so j's response is fixed for fixed δ_j. Suppose that for given δ_i, player i takes action a and that for $\delta_i' = \delta_i + \varepsilon$, $\varepsilon > 0$, player i takes action b. Given r_2 is not a function of δ_i it must be that the reason i switches to b when operating under δ_i' is that it increases the net utility, but results in a loss of more than δ_i from i's greedy best strategy $G_1(\delta_j)$. Given i's utility decreases and the net utility increases, it must be that j's utility increases. □

Corollary 2. *Given a 2-player limited-trust Stackelberg game between a leader i and a follower j, $u_i(\theta_{ij})$ decreases monotonically as δ_i increases.*

Corollary 3. *Given a 2-player limited-trust Stackelberg game between a leader i and a follower j, net utility increases monotonically as δ_i increases.*

Although Lemma 2 shows that for any fixed game, the follower can only benefit if the δ of the leader increases, the same is not always true for the leader if the δ of the follower increases. However, [3] provides strong empirical evidence that in games randomly generated from several types of distributions, the utility of the leader has a strong positive correlation to the δ of the follower. With that in mind, we will make the assumption that given two players l and j such that $\mathcal{A}_{il} = \mathcal{A}_{ij}$, player i would prefer to send an invitation to whichever of the two has a higher δ, and is indifferent between them if $\delta_j = \delta_l$.

Corollary 4. *Given a network G in which all games between any two players have nonnegative expected utilities and are independent, $v_i(\theta)$ is monotonically decreasing with δ_i.*

Proof. Given all games have nonnegative expected utility for i and a game between players i and j is independent of the payoff in a later game between i and l or j and h, all players will accept any games they are invited to. Therefore, by Lemma 2 every term in the sum $v_i(\theta) = \sum_{j \in K_i^1} u_i(\theta_{ij})$ is decreasing monotonically with δ_i and so $v_i(\theta)$ decreases monotonically with δ_i. $\square$

Note that the lemma and corollaries mentioned here do not imply that the utility of the follower j in a particular game decreases monotonically with δ_j, because this is not true: games can be constructed where j's utility increases with δ_j. Intuitively, these games reflect situations in which the leader i can trust j not to take advantage of its strategy s_i, allowing both players to benefit. Once again though, [3] provides empirical evidence that j's *expected* utility decreases monotonically with δ_j for games generated from several distribution types.

Theorem 7. *Given two agents i, j with continuous distributions $\mathcal{A}_{ij}, \mathcal{B}_{ij}$ in which any element d which is dependent on any other set of elements D has a continuous marginal distribution function $f_{d|D}$ for any realization of the elements of D, $u_i(\theta_{ij})$ and $u_j(\theta_{ij})$ are both continuous in δ_i and δ_j provided $u_i(\theta_{ij}), u_j(\theta_{ij}) < \infty$ and have finite variance for all $\delta_i, \delta_j \geq 0$.*

Proof. Consider the set of games $C \subseteq (\mathcal{A}_{ij}, \mathcal{B}_{ij})$ for which $u_i(C, \delta_i, \delta_j)$ is discontinuous on the interval $\delta_i \in [x, x+\varepsilon)$ for $\varepsilon > 0$. Note that in each of these games $(A, B) \in C$, $u_i(A, B, \delta_i, \delta_j)$ is a constant-valued step function where it is not discontinuous. By definition as an expected value,

$$u_i(\theta_{ij}) = u_i(\mathcal{A}_{ij}, \mathcal{B}_{ij}, \delta_i, \delta_j) = \int_{\mathcal{A}_{ij}, \mathcal{B}_{ij}} f_{ij}(A, B) u_i(A, B, \delta_i, \delta_j) dA dB$$

where f_{ij} is the distribution function over $(\mathcal{A}_{ij}, \mathcal{B}_{ij})$. Therefore, as $\varepsilon \to 0$

$$\begin{aligned} & u_i(\mathcal{A}_{ij}, \mathcal{B}_{ij}, x+\varepsilon, \delta_j) - u_i(\mathcal{A}_{ij}, \mathcal{B}_{ij}, x, \delta_j) \\ & = \int_C f_{ij}(A, B) \left(u_i(A, B, x+\varepsilon, \delta_j) - u_i(A, B, x, \delta_j)\right) \end{aligned} \tag{3.1}$$

due to the fact that $u_i(A, B, \delta_i, \delta_j)$ is a constant-valued step function. Note the change in the limits, which comes from the fact that

$$(u_i(A, B, x + \varepsilon, \delta_j) - u_i(A, B, x, \delta_j)) = 0$$

for $A, B \notin C$.

As $\varepsilon \to 0$, then for all $\delta_i \geq 0$ $C \to \varnothing$ by the continuity of the $\mathcal{A}_{ij}, \mathcal{B}_{ij}$ and all marginal distributions therein. Therefore, $u_i(\mathcal{A}_{ij}, \mathcal{B}_{ij}, x + \varepsilon, \delta_j) - u_i(\mathcal{A}_{ij}, \mathcal{B}_{ij}, x, \delta_j)$ goes to 0 because $u_i(\theta_{ij}), u_j(\theta_{ij}) < \infty$ and $u_i(\theta_{ij}), u_j(\theta_{ij})$ have finite variance. Therefore, $u_i(\theta_{ij})$ is continuous in δ_i.

Identical arguments show that $u_i(\theta_{ij})$ is continuous in δ_j, and that $u_j(\theta_{ij})$ is continuous in both δ_i and δ_j. □

In this chapter we assume that $u_i(\theta_{ij}), u_j(\theta_{ij}) < \infty$ and have finite variance for all $\delta_i, \delta_j \geq 0$, all $i, j \in [N]$. We will use Theorem 7 in Section 3.4 for games in which agents are able to change their value of δ.

Having defined the system, we now return to the example we gave earlier in Figure 3.1. Suppose that $k_i = 2$ for all agents i and all games between any players are nonnegative and independent and identically distributed. This means $\mathcal{A}_{ij} = \mathcal{B}_{lh}$ for $i, j, l, h \in [N]$ where $[N] = \{1, 2, ..., N-1, N\}$. Further suppose that $\delta_i < \delta_{i+1}$ for $i \in [6]$; in particular $\delta_i = \frac{2(i-1)}{3}$ for $i \in [7]$. We can then predict exactly who will invite whom to interact (since all outcomes are nonnegative, expected utility for any interaction for both leader and follower is nonnegative and all invitations will be accepted). The behavior is fully characterized by Table 3.3.

Table 3.3: Behavior of Network in Figure 3.1 under Different δ Distributions

		$\delta_i = 2(i-1)/3$			$\delta_i = 2(7-i)/3$		
Player	Degree	Invites	Invited By	Utility per Round	Invites	Invited By	Utility per Round
1	3	5,7	∅	7.750	2,5	2,5,7	15.559
2	4	5,4	∅	7.360	1,3	1,3,4,5	20.446
3	3	5,7	5	11.040	2,5	2,7	12.859
4	4	6,7	2,5,6,7	21.064	2,5	6	9.960
5	4	3,4	1,2,3	15.193	1,2	1,3,4	18.059
6	2	4,7	4,7	13.754	4,7	∅	6.607
7	4	4,6	1,3,4,6	18.931	1,3	6	11.244
Average				13.591			13.559

Table 3.3 presents an interesting interaction between δ and the network structure. In the first set of columns, when $\delta_i = \frac{2(i-1)}{3}$, player 7 is invited to play by every one of its neighbors in the network. This is unsurprising as

$\delta_7 > \delta_{j\neq 7}$. So is player 6, which is again unsurprising as we have $k_i = 2$ for all players $i \in [7]$ and the only player for which $\delta_j > \delta_6$ is $j = 7$. What is more interesting is that player 4 is being invited to play by all of its neighbors, and is engaging in as many games per round as player 7. Further, it is engaging with players that have a higher δ than player 7's partners, both as a leader and as a follower, so we expect that it achieves a higher utility per round than player 7, especially because it is behaving more selfishly. If we define $\mathcal{A}_{ij} = \mathcal{B}_{lh}$ to be a probability distribution over 2×2 matrices with all entries generated from independently and identically from an exponential distribution with $\lambda = 2$, we see that this is exactly what happens. Column 5 represents the average utility each player receives per round after 1000 rounds of play in this setting. Columns 6-8 consider the same setting, but when $\delta_i = \frac{2(7-i)}{3}$, reversing which players are the most valuable partners.

We also remind the reader of the information in Table 3.1 which considers the same values of k and $\mathcal{A}_{ij}, \mathcal{B}_{ij}$. For the two cases in Table 3.3 the average value of δ is 2, so it is unsurprising that they have roughly the same average utility as when δ is uniformly equal to 2 for all agents in Table 3.1. With that being said, the more even distribution of a uniform $\delta = 2$ produces higher average utility. In Section 3.5 we numerically examine the relationship between network structure and δ.

We saw that the behavior of systems with a fixed, known θ can be characterized and readily predicted. We now shift our focus to when parts of θ are unknown or are not fixed. Section 3.3 focuses heavily on the algorithmic methods agents use to learn the δ of their neighbors, and Section 3.4 mathematically details how agents should adjust their δ to maximize their utility in response to their 1- and 2-hop neighborhoods. For those who are concerned primarily with the results of our numerical trials, we recommend skipping ahead to Section 3.5.

3.3 Learning under unknown δ

In this section we consider how agents behave when they don't have accurate information about their their neighbors' δ values. The expected utility an agent i will gain by partnering with another agent j is dependent on both the utility j brings ($\mathcal{A}_{ij}$ and $\mathcal{B}_{ij}$) and its trustworthiness (δ_j). These parameters are independent, allowing each to be estimated separately. $\mathcal{A}_{ij}, \mathcal{B}_{ij}$ are multidimensional distributions and thus can be estimated using standard statistical methods such as kernel density estimates and expected log-likelihood if they are not known *a priori*. Therefore we focus on how agent i estimates δ_{-i} from observations, where $\delta_{-i} = \{\delta_j\}_{j \in [N] \setminus \{i\}}$ is the set of δ values for all agents other than i.

Note that as much of this section considers only interactions between two players in a game rather than agents in a larger network, we will use the terms "player" and "agent" interchangeably here.

3.3.1 Learning δ_{-i} as Leader

Consider an $m \times n$ leader-follower game with leader player 1 and follower player 2. Assume that player 1 knows through past observations that $\delta_2 \in [\delta_{21}^l, \delta_{21}^u)$, a pair of lower and upper bounds. The interval $[\delta_{21}^l, \delta_{21}^u)$ is half-open because when we observe δ_{ji}^l being given up we know $\delta_j \geq \delta_{ji}^l$, but when δ_{ji}^u is not given up all we know is $\delta_j < \delta_{ji}^u$. Suppose that player 1 has selected strategy s_i to play. At this point, the game is equivalent to the $1 \times n$ game given in Table 3.4.

Table 3.4: $1 \times n$ Leader-Follower Game

		Player 2		
		s_1	...	s_n
Player 1	s_i	(a_1, b_1)	...	(a_n, b_n)

Player 1 can refine its knowledge of δ_2 based on player 2's response by considering the Pareto frontier of player 2's strategies measured in the values of u_2 and $u_1 + u_2$. Without loss of generality, assume that there are k strategies on the frontier and they are relabeled $\{s_1, s_2, ..., s_k\}$ such that $u_2(s_1) > u_2(s_2) > ... > u_2(s_k)$ and $u_1(s_1) + u_2(s_1) < u_1(s_2) + u_2(s_2) < ... < u_1(s_k) + u_2(s_k)$. Figure 3.2 gives an example of such a frontier with $k = 5$. If player 2 plays s_j in response, then it must be that $b_1 - b_j \leq \delta_2 < b_1 - b_{j+1}$. Let $b_{k+1} = -\infty$ for the case where $j = k$.

Analyzing the Pareto frontier allows player 1 to determine whether a better bound for δ_2 is found and whether δ_2 has changed. Consider Figure 3.2 again: based on player 1's previously derived bounds for δ_2, it expects player 2 to

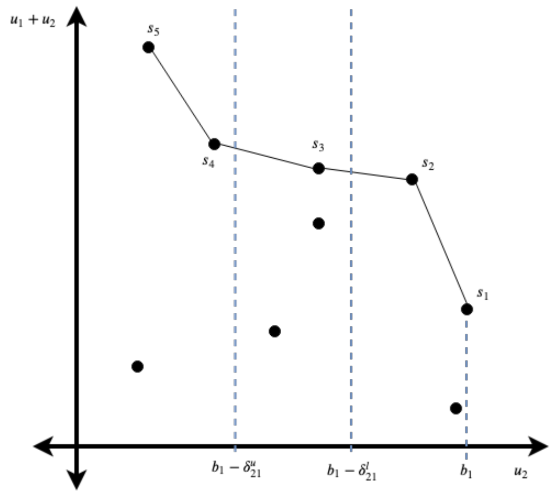

Figure 3.2: Pareto frontier of game/strategy in Table 3.4 for the Leader.

select either s_2 or s_3 in response, depending on whether $b_1 - b_3 > \delta_2$. This means that if player 2 responds with $s_j \in \{s_1, s_4, s_5\}$, there has been a change to δ_2. Algorithm 1 gives the full method for the leader player 1 to update its bounds for the follower player 2, while noticing any detectable changes in δ_2.

Algorithm 2 Leader: Update $\delta_{21}^l, \delta_{21}^u$

Require: $\delta_{21}^l, \delta_{21}^u, S = \{s_1, s_2, ..., s_k\}, j$
  $l \leftarrow b_1 - b_j$
  $u \leftarrow b_1 - b_{j+1}$
  $change \leftarrow False$
  if $l \geq \delta_{21}^u$ **or** $u \leq \delta_{21}^l$ **then**
    $\delta_{21}^l \leftarrow l$
    $\delta_{21}^u \leftarrow u$
    $change \leftarrow True$
  else
    $\delta_{21}^l \leftarrow \max\{l, \delta_{21}^l\}$
    $\delta_{21}^u \leftarrow \min\{u, \delta_{21}^u\}$
  end if
  return $\delta_{21}^l, \delta_{21}^2, change$

3.3.2 Learning δ_{-i} as Follower

We now show how a follower can learn the δ value of a leader. Consider an $m \times n$ game with leader player 1 and follower player 2. Assume that player 2 knows from past observations that $\delta_1 \in [\delta_{12}^l, \delta_{12}^u)$. Recall that player 2's best response function to player 1 selecting s_i is $s_j^* = r_2(s_i, \delta_2)$. While

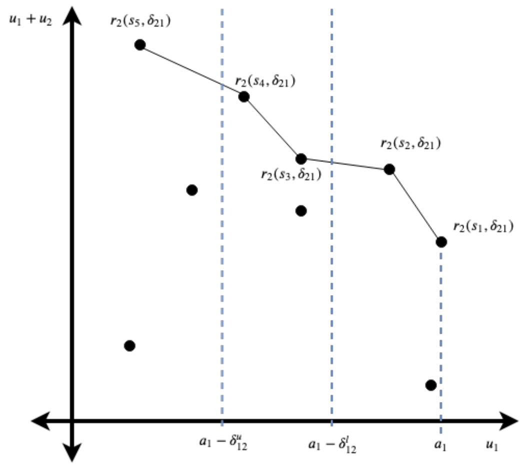

Figure 3.3: Pareto frontier of game in Table 3.5 for the Follower

player 1 does not know δ_2, it does have an estimate δ_{21}. Unless otherwise specified, such as if player 1 has some additional *a priori* knowledge about δ_2, $\delta_{21} = \frac{\delta^u_{21}+\delta^l_{21}}{2}$ the mean of the observed bounds. Thus from player 1's perspective, the game can be rewritten as the $m \times 1$ game in Table 3.5.

Table 3.5: $m \times 1$ Leader-Follower Game

		Player 2
		$r_2(s_i, \delta_{21})$
Player 1	s_1	(a_1, b_1)
	$\vdots$	$\vdots$
	s_m	(a_m, b_m)

Provided that player 2 knows δ_{21}, player 1's estimate of δ_2, it is able to construct the $m\times 1$ game that player 1 is considering. Based on past interactions, player 2 can compute $\delta^l_{21}, \delta^u_{21}$ exactly from its own past actions, and therefore compute δ_{21}. Player 2 can then construct a Pareto frontier of player 1's strategies similar to Figure 3.2, but measured in the values of u_1 and u_1+u_2. Without loss of generality, assume that if there are k strategies on the frontier, they are relabeled $\{s_1, s_2, ..., s_k\}$ such that $u_1(s_1) > u_1(s_2) > ... > u_1(s_k)$ and $u_1(s_1)+u_2(s_1) < u_1(s_2)+u_2(s_2) < ... < u_1(s_k)+u_2(s_k)$. Figure 3.3 gives an example of such a frontier with $k = 5$. If player 1 selects s_i while anticipating $r_2(s_i, \delta_{21})$ in response, it must be that $a_1 - a_i \leq \delta_1 < a_1 - a_{i+1}$, where $a_{k+1} = -\infty$ in the case that $i = k$. δ^l_{12} and δ^u_{12} are updated if this implies a better bound.

Similar to the leader in the previous section, the follower can also use the Pareto frontier to determine if δ_1 has changed. For the frontier in Figure 3.3, player 2's previously derived bounds for δ_1 indicate that if δ_1 hasn't changed, player 1 will select s_2, s_3, or s_4, with s_2 occurring if $\delta_1 < a_1 - a_3$. This means if player 1 selects s_1 or s_5, there has been a change to δ_1. The follower can then use Algorithm 1 with slight modifications (consider b_1, b_i, b_{i+1} rather than a_1, a_i, a_{i+1}) to update its knowledge of the leader player 1, while noticing any detectable changes in δ_1.

3.3.3 Network Dynamics Under Unknown δ

So far in this section we have focused on the learning of unknown δ between two players. Now we turn our attention to the network as a whole. Recall that each agent in the network can initiate at most k interactions per round. Therefore, any agent i with neighborhood $|N_i^1| > k_i$ faces an exploration-exploitation dilemma for each of its invitations: select a neighbor j for which $u_i(\theta_{ij})$ is maximized or select a neighbor h for which $\delta_{hi}^u - \delta_{hi}^l$ is large.

We consider agents which address this dilemma in the following manner: At the beginning of round t, agent i selects at most k_i other agents from its neighbors N_i^1 to interact with. At the beginning of each round, agent i selects some of these neighbors for the purpose of exploration and some for exploitation. For each neighbor j, although i does not know δ_j it has an estimate δ_{ji} based on past interactions. Unless otherwise specified, $\delta_{ji} = \frac{\delta_{ji}^u + \delta_{ji}^l}{2}$. If i decides to exploit $h(t)$ of its interactions in round t, then it selects the set of S neighbors such that $S = \arg\max_S \sum_{j \in S} u_i(\theta_{ij}), \quad \text{subject to } |S| \leq h(t)$ and $u_j(\theta_{ij}) \geq 0 \ \forall j \in S$, as otherwise the invitation will not be accepted. Note that based on past interactions, player i is capable of determining the value δ_{ij} that player j estimates for δ_i, and so can avoid sending an invitation which is likely to be rejected. $h(t)$ is determined according to a multi-armed bandit scheduling policy, such as uniform ε-greedy, -first, or -decreasing, or a more sophisticated policy such as Thompson sampling. Player i then randomly samples $k_i - |S|$ of its remaining neighbors for exploration, according to a discrete probability distribution $f_t(N_i^1 \setminus S, \delta_i')$, where $\delta_i' = \{(\delta_{ji}^l, \delta_{ji}^u)\}_{j \in N_i^1}$ is i's estimates of the δ values of its neighbors. For each invited neighbor j player i then plays according to δ_i and its estimate δ_{ji}: player i should not forego a large opportunity or accept a large cost while interacting with a neighbor it is exploring. Instead, it will take that into account in determining the value of issuing j future invitations.

3.4 Network Games with Variable δ

In Sections 3.2 and 3.3 we defined the basic mechanics under which network games function. A natural extension of this system is considering how a player i might change δ_i in order to take advantage of $\delta_{-i} = \{\delta_1, \delta_2, ..., \delta_{i-1}, \delta_{i+1}, ..., \delta_N\}$. Therefore, if agent i can change δ_i between rounds, it should set it to

$$\delta_i^* = \arg\max_{\delta_i \in \Delta_i} \mathbf{u}_i(\theta_{-i}),$$

where $\theta_{-i} = \{G, \mathcal{A}, \mathcal{B}, \delta_{-i}\}$ and where $\Delta_i = [0, \delta_{max}]$ and δ_{max} is an arbitrarily enforced maximum value of δ for the system. Note that while δ_{max} can be arbitrarily large, we require $\delta_{max} \in \mathcal{R}^+$.

Finding δ_i^* is complicated by two factors. The first is that agent i does not know δ_{-i}. Instead, it has estimates of δ_{ji} for its neighbors $j \in N_i^1$ based on past interactions. This prevents it from accurately determining whether or not it will receive an invitation from j for a given value of δ_i, and it will be forced to estimate the expected value of the game if it does receive the invitation. The second complication is that agent i's neighbor j does not know δ_{-j}: agent j will not immediately notice a change in δ_i, and so agent i will not immediately receive the expected utility from shifting from δ_i to δ_i^*. This occurs in the form of j not correctly deciding whether or not to issue an invitation to i, as well as not estimating agent i's trust level correctly if they do interact.

With mild re-use of notation, we will let $\delta_i' = \{\delta_{ji}\}_{j \in N_i^1}$ be agent i's estimates of δ_j for each of its neighbors j. Because we are considering social networks, it is reasonable to assume that agents share knowledge and impressions of their own neighbors during their interactions via gossip. This leads to agent i knowing δ_j' for each of its neighbors $j \in N_i^1$. i can thus address the first complication in finding δ_i^*: by knowing δ_j' for each of its neighbors j, agent i can predict whether or not it will receive an invitation from agent j if it shifts the value of δ_i. As noted in Section 3.2, agent i's expected utility can be determined entirely from its 2-hop neighborhood. i therefore does not need any additional information from other agents $l \notin N_i^1$. While it must still estimate the expected value of the game that agent j initiates as a function of δ_{ji} and δ_i, after several interactions it is likely that $\delta_{ji} \approx \delta_j$. By the continuity implied by Theorem 7, this means that as $\delta_{ji} \to \delta_j$, $u_i(\mathcal{A}_{ji}, \mathcal{B}_{ji}, \delta_{ji}, \delta_i) \to u_i(\theta_{ji})$ and $u_i(\mathcal{A}_{ij}, \mathcal{B}_{ij}, \delta_i, \delta_{ji}) \to u_i(\theta_{ij})$. This addresses the first complication in determining δ_i^*.

The second complication is addressed heuristically. Because it takes time for knowledge of a change in δ_i to become apparent to agent i's neighbors $j \in N_i^1$, particularly if the change is small, it is not to i's benefit to change δ_i frequently. Doing so will result in it never gaining the expected utility it computed when determining δ_i^*. Therefore, agent i will only recompute δ_i in a given round with arbitrary probability p_i. The number of rounds that each agent i will commit to a given δ_i before reevaluating then becomes a geometric random variable. Another heuristic option is for agents to update on an epoch schedule, after every t rounds. This allows agent i to better estimate δ_i', and allows agent $j \in N_i^1$ to better determine δ_i so that i realizes the value it expected when it set δ_i. We consider both of these mechanisms in our numerical trials in Section 3.5.

Having covered the mechanics by which δ_i varies for a given agent i, we now address changes to the exploration-exploitation methodology given in Section 3.3.3. This framework functioned well when δ was fixed for all agents, as the value of exploration decreased with accumulated knowledge. However, whenever δ_j shifts, agent i's past knowledge of δ_j becomes obsolete and additional exploration is beneficial. This means that rather than using $h(t)$ to determine how many of its neighbors to exploit and explore, agent i should now also take into account how recently each of its neighbors $j \in N_i^1$ changed δ_j. This is easily accomplished, as Algorithm1 1 reports whether or not a change in δ_j has been detected. Therefore, a vector t_i of how many rounds ago each neighbor $j \in N_i^1$ changed δ_j can easily be maintained and exploitation and exploration can instead be determined by $h(t_i)$.

3.4.1 Personalized δ

Before beginning numerical studies in the next section, we first consider how the system functions if an agent i uses different values of δ_i depending on which of its neighbors it is interacting with. This reflects the fact that some individuals may prefer specific trusted partners or be more willing to help them than they would other acquaintances.

Let $\delta_i(j)$ be the value of δ_i agent i uses when interacting with agent j. Similarly, let $\delta_{ji}(i)$ be agent i's estimate of $\delta_j(i)$. In many ways this will make the problem of selecting δ_i^* simpler, despite the fact that agent i now needs to select a vector rather than a single value. This is because i can determine $\delta_i^*(j)$ while only considering j and N_j^1, and only taking N_i^1 into account at the end.

The procedure for agent i to determine $\delta_i^*(j)$ is straightforward. First, i determines $\delta_i^F(j)$, where

$$\delta_i^F(j) = \arg\max_{\delta_i(j)\geq 0} I_1(\mathcal{A}_{ji}, \mathcal{B}_{ji}, \delta_j(i), \delta_i(j)) \max\{0, u_i(\mathcal{A}_{ji}, \mathcal{B}_{ji}, \delta_j(i), \delta_i(j))\}$$

and $I_1(\mathcal{A}_{ji}, \mathcal{B}_{ji}, \delta_j(i), \delta_i(j))$ is an indicator function which is 1 if agent j will issue i an invitation and 0 otherwise. $\delta_i^F(j)$ is the optimal value of $\delta_i^*(j)$ if agent i does not intend to issue j an invitation, but would still benefit from receiving one from j. Note that the term $\max\{0, u_i(\mathcal{A}_{ji}, \mathcal{B}_{ji}, \delta_j(i), \delta_i(j))\}$ indicates that i will decline the invitation if $u_i < 0$. Next, agent i determines $\delta_i^L(j)$ such that

$$\begin{aligned}\delta_i^L(j) = \arg\max_{\delta_i(j)\geq 0} & I_1(\mathcal{A}_{ji}, \mathcal{B}_{ji}, \delta_j(i), \delta_i(j)) \max\{0, u_i(\mathcal{A}_{ji}, \mathcal{B}_{ji}, \delta_j(i), \delta_i(j))\} \\ & + I_2(u_j(\mathcal{A}_{ij}, \mathcal{B}_{ij}, \delta_i(j), \delta_j(i)) \geq 0) u_i(\mathcal{A}_{ij}, \mathcal{B}_{ij}, \delta_i(j), \delta_j(i))\end{aligned}$$

where I_2 is an indicator variable which is 1 if agent j would accept an invitation from agent i at the specified $\delta_i(j)$. $\delta_i^L(j)$ represents the optimal value of $\delta_i^*(j)$ if agent i would like to issue an invitation to j, as well as potentially receive one.

Having compiled a pair $(\delta_i^F(j), \delta_i^L(j))$ for each neighbor $j \in N_i^1$, agent i now can select $\delta_i(j)$ as one of the two values from each pair. This is subject only to the constraint that i may select $\delta_i(j) = \delta_i^L(j)$ for at most k_i of its neighbors, as it can only issue at most k invitations. By choosing at most k_i

neighbors for which this difference is highest, agent i can determine δ_i^*.

Note that while we assumed δ_{-i} to be known for simplicity, the procedure for determining $\delta_i^*(j)$ when δ_{-i} is unknown is analogous, utilizing the techniques from earlier in Section 3.4.

3.4.2 Variable Known δ

Before moving on we again pause here to consider the case when δ_{-i} is known to agent i. While agents in the network interact with each other in 2-player Stackelberg games, allowing agents to modify their values of δ between rounds gives the agents a second, indirect way to interact with each other. When each agent selects a value for δ it does not directly result in utility for the agent, but it influences which agents will interact with it as well as how they will interact, which in turn results in utility. For this reason, selecting δ represents an N-player game on top of the system. Because the strategies for this game indirectly influence utility, instead influencing the system that will determine utility, we refer to this as an N-player continuous strategy "metagame" over the system. As each player displays trust only in its own self-interest, to attract more interactions, we can show that under certain settings the metagame displays mixed Nash equilibria through use of [57]'s results for Nash equilibria in games with continuous strategy sets.

Definition 9 (Mixed Nash Equilibrium)**.** *Given an N-player game with strategy profiles $\sigma = (\sigma_1, \sigma_2, ..., \sigma_N)$ for each player where for a given player i, σ_{-i} is the set of strategies played by all other players, σ is a mixed Nash equilibrium (MNE) if and only if for any other valid strategy profiles σ_i', $u_i(\sigma_i, \sigma_{-i}) \geq u_i(\sigma_i', \sigma_{-i})$ for all $i \in [N]$, and $u_i(\sigma_i, \sigma_{-i})$ is the expected utility of the game for player i when it plays strategy σ_i.*

Theorem 8. *Consider a social network G with uniform interactions $\mathcal{A}_{ij} = \mathcal{B}_{lh}$ for all $l, i, j, h \in [N]$ such that all payoffs are nonnegative and for agent i with neighbors j and l, $\delta_j \leq \delta_l \to u_i(\theta_{ij}) \leq u_i(\theta_{il})$. Then the N-player metagame with closed interval strategy space $\Delta_i \subseteq \mathcal{R}$ and utility function $\mathbf{u}_i$ for $i \in [N]$ possesses a mixed Nash equilibrium.*

The proof to Theorem 8 is given in Appendix B.2.1.

3.5 Numerical Studies

In this section we empirically examine the relationship between network position and trust level δ in real social networks. We will study Zachary's Karate Club network from [58], as well as the ego-Facebook network curated by SNAP. The karate club network is visually represented in Figure B.7 in Appendix B.3. We will consider separately when δ_{-i} is known and unknown for agent i in each network. When agent i determines which of its neighbors $j \in N_i^1$ to issue invitations to, $u_i(\theta_{ij})$ is estimated as the mean of i's utility in 1000 games drawn independently and identically at random from $\mathcal{A}_{ij}, \mathcal{B}_{ij}$ with δ_i, δ_j. We will confine our attention to distributions where $u_j(\theta_{ij}) \geq 0$, so that all invitations will be accepted.

3.5.1 Known δ

We consider the Karate Club network with the following parameters:

- $\mathcal{A}_{ij} = \mathcal{B}_{hl}$ for all $h, i, j, l \in [N]$. $A_{ij} \sim \mathcal{A}_{ij}$ is a 2×2 matrix with entries generated independently and identically from the exponential distribution with $\lambda = 4$.
- $\delta_i \in [0, 30]$ $\forall i \in [N]$.
- δ_i is known to all agents.
- δ_i updates between rounds. δ_i at time t is a greedy best response to δ_{-i} at time $t-1$. For $t = 0$, $\delta_i = 0$, $\forall i \in [N]$.
- $k_i = 2$ for each agent i.

$u_i(\theta_{ij})$ and $u_j(\theta_{ij})$ as functions of δ_i, δ_j are estimated by taking the sample mean utilities of 1000 simulated games generated independently and identically according to $\mathcal{A}_{ij}, \mathcal{B}_{ij}$.

Figure 3.4 illustrates $\mathbf{u}_1(\theta_{-1}, \delta_1)$ as a function of δ_1 for fixed (random) δ_{-i}, for vertex 1 in the karate club network. The left plot gives the value of $\mathbf{u}_1(\theta_{-1}, \delta_1)$ and the right plot shows the number of games the agent at vertex 1 in the Karate Club network participates in per round. We see in the left plot that $\mathbf{u}_1(\theta_{-1}, \delta_1)$ is monotonically decreasing mildly with δ_1 at all but a

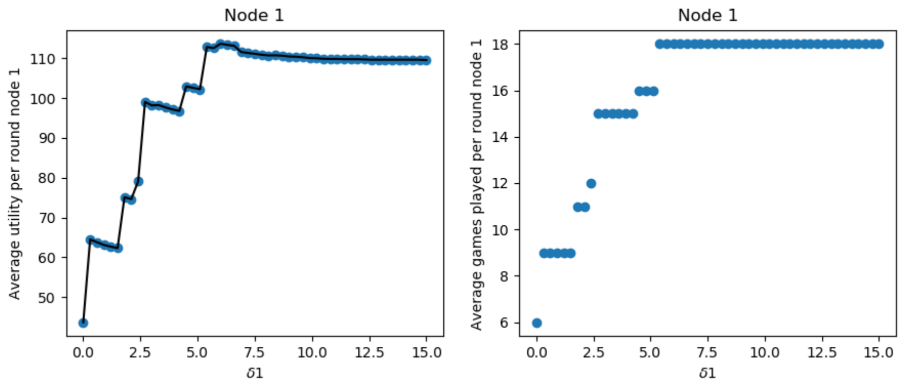

Figure 3.4: Utility for Vertex 1, Known δ

handful of points where there is a sharp increase. These points occur when δ_1 is large enough to attract a new player to begin interacting with agent 1, as shown by the plot on the right. This is unsurprising: the utility of a single game for the follower may not monotonically decrease as δ increases, but as stated in Section 3.2 we strongly suspect that for many distributions it monotonically decreases *in expectation*.

Now we examine how players behave under the parameters above when all are adjusting δ together between rounds. As noted, $u_i(\theta_{ij})$ is estimated numerically. Between rounds, it is sampled at a number of points in the interval $[0, 30]$ then set to the one which maximizes $\mathbf{u}_i(\theta_{-i}, \delta_i)$. Each curve in Figure 3.5 plots δ_i for an individual agent as it varies over time. We see the majority of players reach $\delta_{max} = 30$ and stay there. They do so in order to be competitive in attracting partners, with only occasional decreases to $\delta = 0$ when they are not competitive enough. The curves in Figure 3.6 display the mean value of δ for all verteces of the same degree. For example, the yellow curve is the mean value of δ across all verteces of degree 4. Figure 3.6 clearly shows what is occurring: we see that at any given time there are generally between 0 and 2 verteces of degree 2 (out of 11) with low values of δ. The variance of which ones are not at $\delta = 30$ at any time may be due to best response dynamics between them, or to insufficient sample size when estimating utilities.

Vertex 12, the only vertex with degree 1, is easier to examine. Its only neighbor is vertex 1, which has degree 16. If all of vertex 1's other neighbors have $\delta = 30$, then if $\delta_{12} = 30$, vertex 12 can expect to be invited to $\frac{1}{8}$ games per round. Figure 3.6 indicates that $u_{12}(\theta_{12,1}, \delta_{12} = 0) > u_{12}(\theta_{12,1}, \delta_{12} =$

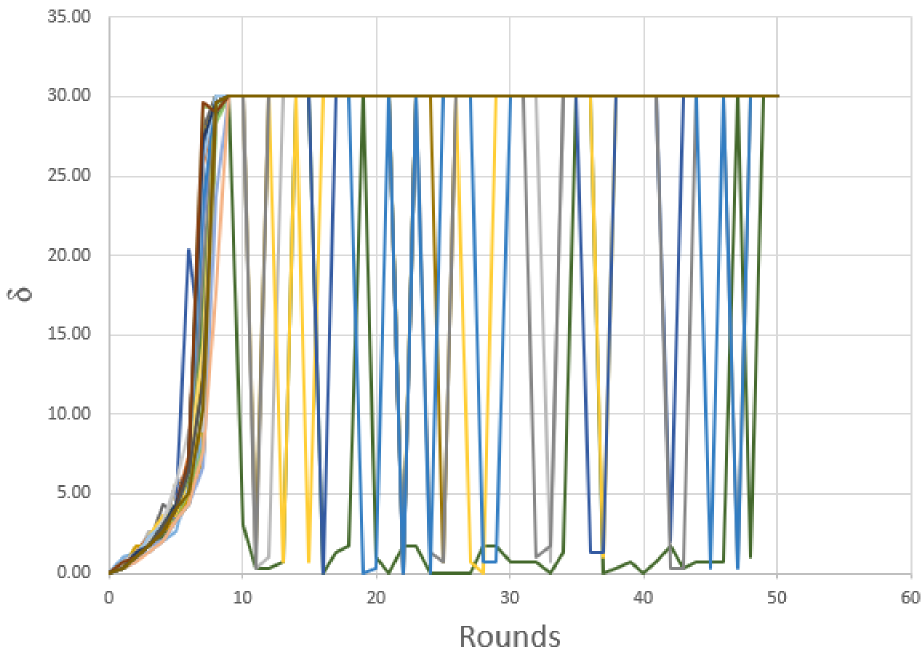

Figure 3.5: δ in karate club network with $k = 2$ invitations per round with ties broken uniformly at random

$30) + \frac{1}{8}u_{12}(\theta_{1,12}, \delta_{12} = 30)$. The occasional jumps of $\delta_{12} = 30$ can be attributed to the changes in behavior of the verteces of degree 2. The points at which δ_{12} is low but not 0 are attributable to insufficient sample size.

We previously focused on ties being broken randomly when deciding whom to issue invitations to. Now, we examine when ties are broken according to the lexicographic ordering. Figure 3.7 is analogous to Figure 3.5 with this change. It shows a strong, consistent, and repeating pattern in the values of player δs over time. When δ is low for most players, there is a general pattern of one-upsmanship between players: Each tries to slightly outdo its competitors, which progresses toward large jumps as the costs of increasing δ relative to a player's current δ value shrink. The resulting pattern is a very clear S-curve. However, once many players reach $\delta = 30$ the "losers" of the tie-breakers drop down to $\delta = 0$ resulting in a gradual cascade of all players back to low values of δ, at which point the process repeats. This echoes the findings of [39], in which players "fake" generosity to attract partners: once they have either failed to attract partners or their competitors have given up, each player returns to selfish behavior until it is once again forced to behave in a trustworthy manner due to competition. In contrast, randomly

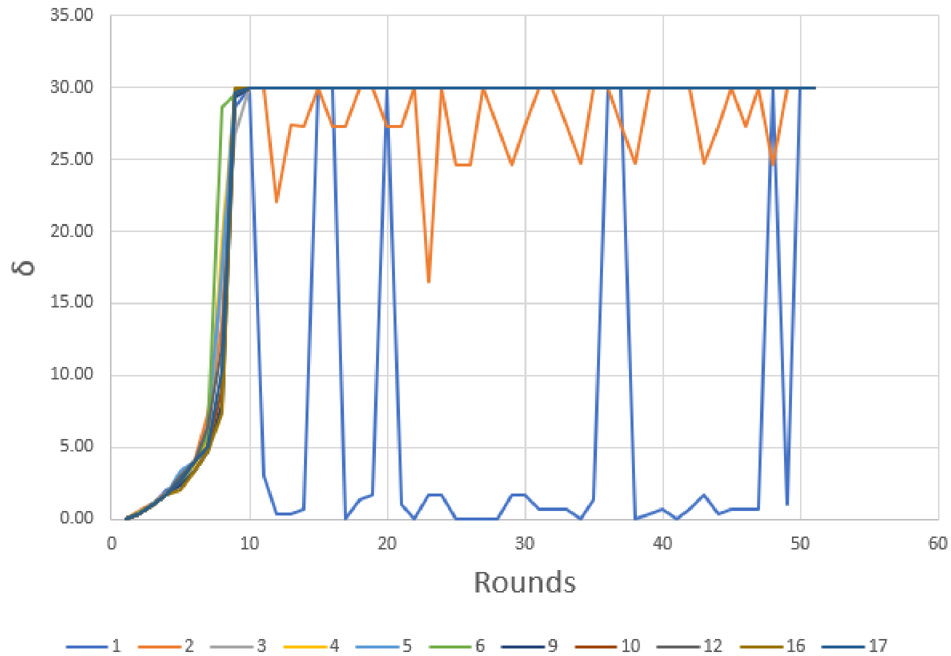

Figure 3.6: Mean δ by vertex degree in karate club network with $k = 2$ invitations per round with ties broken uniformly at random

breaking ties maintains a constant state of competition, stopping backsliding. Further, it is supported by [56] which finds that semi-frequent partner changes are necessary to motivate generous behavior, as otherwise partners become complacent and attempt to take advantage of each other.

We do note that we suspect that this behavior in which all agents cyclically return to low values of δ does not persist when agents are less myopic. To demonstrate that, we consider agents who make updates to their δ value by maximizing the sum of their expected utility and a constant $\rho > 0$ times their expected utility after their neighbors react to the new value of δ. Figure 3.8 considers this setting with verteces conjoined by degree in the karate club network with lexicographic tie-breaking and $k = 2$ invitations per player with $\rho = 0.8$. Due to increased computation, expected utility is determined as the mean of 200 games drawn independently and identically at random rather than 1000. While Figure 3.8 still shows evidence of some cyclic behavior, it appears greatly reduced and more stable than that in Figure 3.7. This leads us to suspect it would be further reduced by even more far-sighted agents, however computational concerns prevent us from testing this.

We also consider a second, larger social network where $N = 333$. This

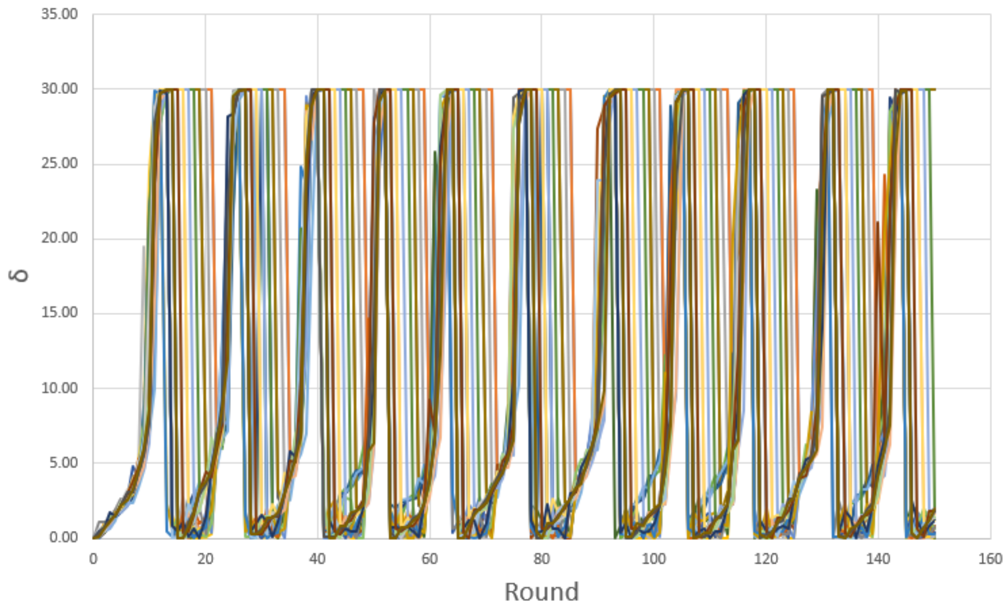

Figure 3.7: δ in karate club network with $k = 2$ invitations per round with lexicographic tie-breaking.

network is a subset of the ego-Facebook network curated by SNAP. We let $k_i = 3$ invitations for each agent i and keep all other conditions identical to our previous setting, breaking ties uniformly at random. Figure 3.9 displays the same behavior which occurred in the Karate Club network in Figure 3.5, rather than the repeated S-curves seen in Figure 3.7. Similarly, nearly all agents stayed at $\delta_{max} = 30$ with a small number playing very small values of $\delta \approx 0$. This divide, with approximately $\frac{5}{6}$ agents using a high δ while $\frac{1}{6}$ us a low δ, leads to a remarkably stable average $\delta \approx 25$. Also as in Figure 3.5, there are a handful of agents which occasionally drop to low values of δ for a short time period when they judge it too competitive before returning to $\delta = 30$. This suggests that the pattern of an S-curve increase in δ followed by a plateau may be characteristic of naturally occurring social networks engaged in partner selection.

3.5.2 Unknown δ

In the Section 3.5.1 we considered networks in which each player i knew the value of δ_{-i}, and was immediately aware of any changes in it. Now we consider behavior when δ_{-i} is unknown, and agent i estimates it using Algorithm 1.

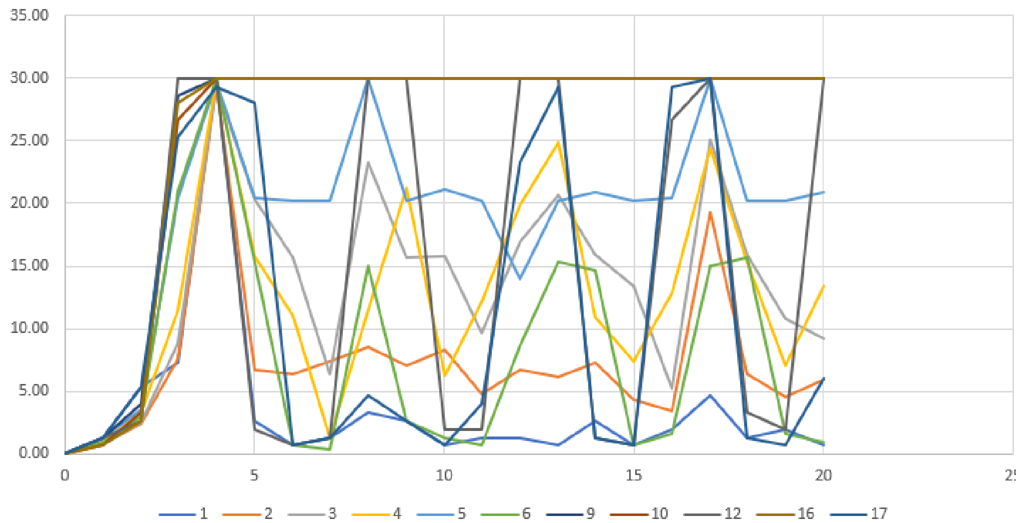

Figure 3.8: Mean δ by vertex degree in karate club network with $k = 2$ invitations per round with ties broken lexicographically, non-myopic updates

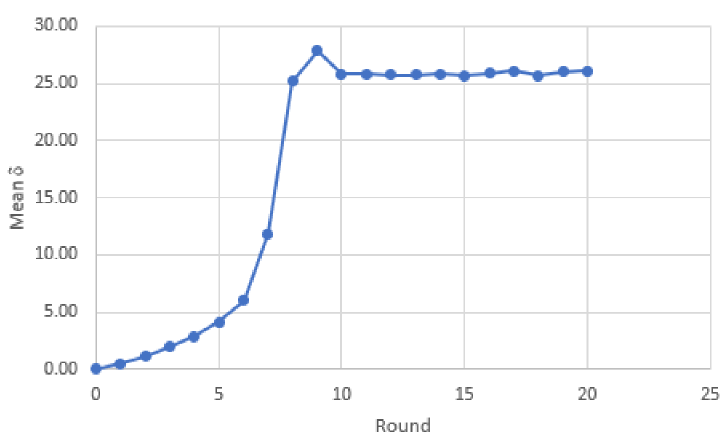

Figure 3.9: Mean δ in Facebook Ego Network

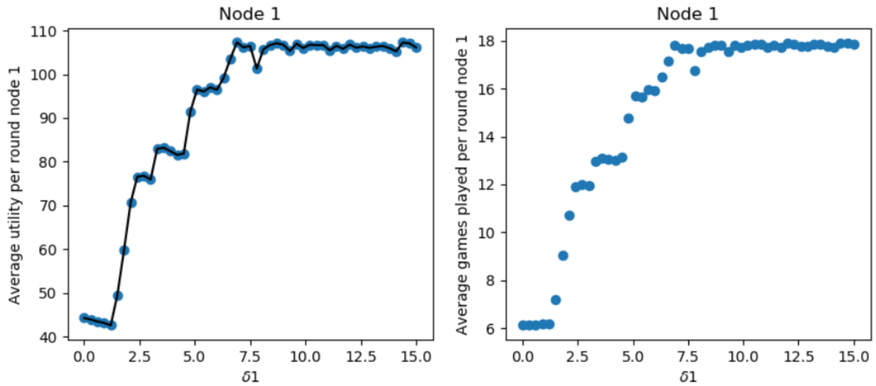

Figure 3.10: Utility for Vertex 1, unknown δ_{-1}

We again consider the Karate Club Network and ego-Facebook Network. All parameters will be identical to those in the previous subsection for ease of comparison. The points in the curve in Figure 3.10 are computed as the average of 1000 independent identically distributed games and provides a direct comparison to Figure 3.4. We see that the estimates in the unknown case approximate those in the known case. This suggests that as knowledge of δ_{-i} improves, players will approach the same behaviors they display when δ_{-i} is known to agent i.

We consider agents who update their values of δ in the two ways we discussed in Section 3.4: all agents either update their δ value probabilistically between rounds, or all agents work on an epoch system, updating their δ values every t rounds after they and their neighbors have learned about each other.

Figure 3.11 illustrates how delta shifts when players update according to an epoch system, every $t = 100$ rounds. The acceleration to $\delta_{max} = 30$ is faster than in the known δ setting, likely due to agents overestimating the trustworthiness of those they are competing with and accidentally overcompensating in response. Due to the fact that players only have estimates of each other's δ, we see some of the gaming which occurs in Figure 3.7: agents attempt to lower their δ value once they believe they've discouraged their competitors down to lower δ values. Upon learning otherwise they increase again. However, the time these agents spent with decreased δ values acts as a signal to their competitors. These competitors then believe that they can lower their δ value in the same way the original agent did, acting as a new

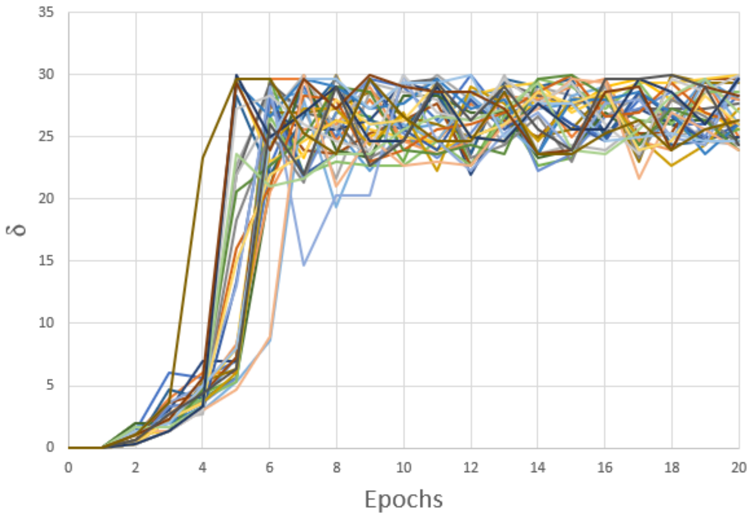

Figure 3.11: δ in Karate Club Network, epoch = 100 rounds, $k = 2$ invitations per round

signal for the original agent. This cycle leads to an oscillation between points in the range $[23, 30]$ for δ. However, agents' estimates of their competitors are accurate enough that they rarely underestimate them badly enough to drop below this range.

Figure 3.12 illustrates the same setting when players independently update their δ value with probability $\frac{1}{100}$ after each round. The behavior is very similar to that in the previous figure, again showing the attempts at gamesmanship where agents try to lower δ once they feel they have discouraged competition. They also oscillate within a similar range of values for δ. It is worth noting that for both heuristic update schemes, lexicographic tie-breaking results in identical behavior to that pictured in Figures 3.11 and 3.12 rather than the sharply cyclical behavior seen in Figure 3.7. This is likely due to agents not having any ties to break, as the continuous payoff distribution leads to $\delta_{ji} \neq \delta_{li}$ for $l, j \in N_i^1$, even if $\delta_j = \delta_l$.

Finally, we consider the ego-Facebook network with $k_i = 3$ invitations for each agent i. The mean δ value is displayed in Figure 3.13, where players update according to the epoch system. The comparison between Figures 3.9 and 3.13 appears identical to the Figures 3.5 and 3.11. As in Figure 3.13, the network quickly settles to an average δ value of ≈ 25. Unlike in that figure, player behavior is closer to that of Figure 3.11, with the majority of players oscillating in the range [23,30], with some few consistently playing $\delta \approx 0$.

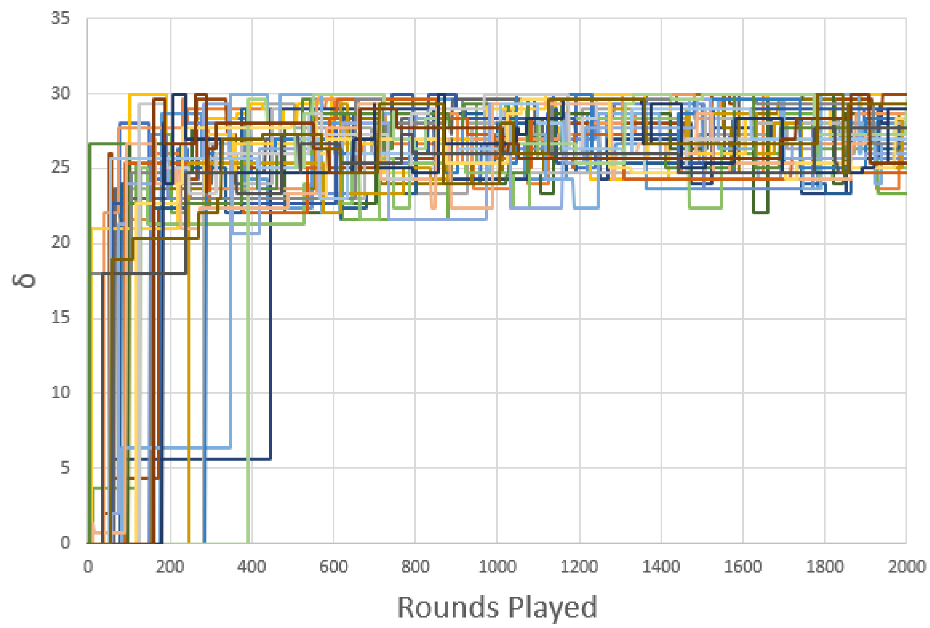

Figure 3.12: δ in Karate Club Network with update probability $= \frac{1}{100}$, $k = 2$ invitations per round

We see in both networks that competition between agents strongly pushes them to maintain increasing levels of trustworthiness; this stops only at the point they are no longer competitive. This increase in trustworthy behavior is healthy for the system as a whole both when δ is known and when it is not. When $\delta = 0$ for all players in the karate club network under the setting considered, the average utility per player per round is 22.792. When δ is known it is 27.578, when δ is unknown and updates are on an epoch schedule it is 27.540, and when δ is unknown and updates probabilisticly it is 27.747. Similarly for the ego-Facebook network, when $\delta = 0$ the average utility per player per round is 30.905; when δ is known it is 37.750 and when δ is unknown and updates on an epoch schedule it is 38.297.

3.6 Discussion

In the previous section, there was an increase in trustworthiness which occurred naturally in both social networks. However, we can construct "artificial" networks in which this does not occur. Consider the 5-star graph in Figure 3.14: the central vertex does not need to compete to receive in-

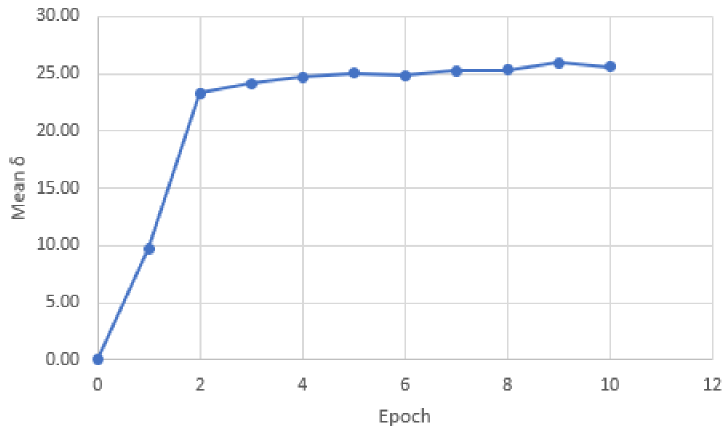

Figure 3.13: Mean δ in Facebook Ego Network, epoch = 100 rounds, $k = 2$ invitations per round

vitations from the other verteces, as it is their only option. Conversely, for $k_i \leq 4$ for the central vertex i, the other verteces must compete to attract the invitations of the central vertex, and will be forced to maintain a high level of trustworthiness. For $k_i \geq 5$ this competition goes away: each non-central vertex will receive an invite provided the expected utility for the central vertex is positive. In this case all δ values drop to 0. This is an interesting dynamic, and one which is at play in all networks: by restricting a resource, in this case the number of invitations which may be issued, agents actually become more trustworthy in their dealings with each other as behaving otherwise causes a loss of access to the resource. Counter-intuitively, they thus become more selfish when the resource is abundant, rather than less. This is particularly apparent in the diad graph in Figure 3.15: each player knows that it is the other players only option and thus has no need to compete to attract an invitation. However, the fact that these players do not need to compete does not mean that both do not stand to benefit from trustworthy behavior in the long term, merely that it is no longer an attractive option for myopic agents focused only on the short term. Traditional mechanisms for repeated games such as as grim trigger and discounted horizon analysis have been shown to help players avoid such avoid such short-sighted behavior, which is why we hope to incorporate these techniques into future versions of our system.

Nonetheless, as we noted previously this behavior does not seem to exist in naturally occurring social networks. While there are "singleton" verteces in

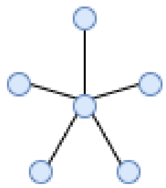

Figure 3.14: A 5-star graph

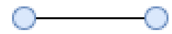

Figure 3.15: A diad graph

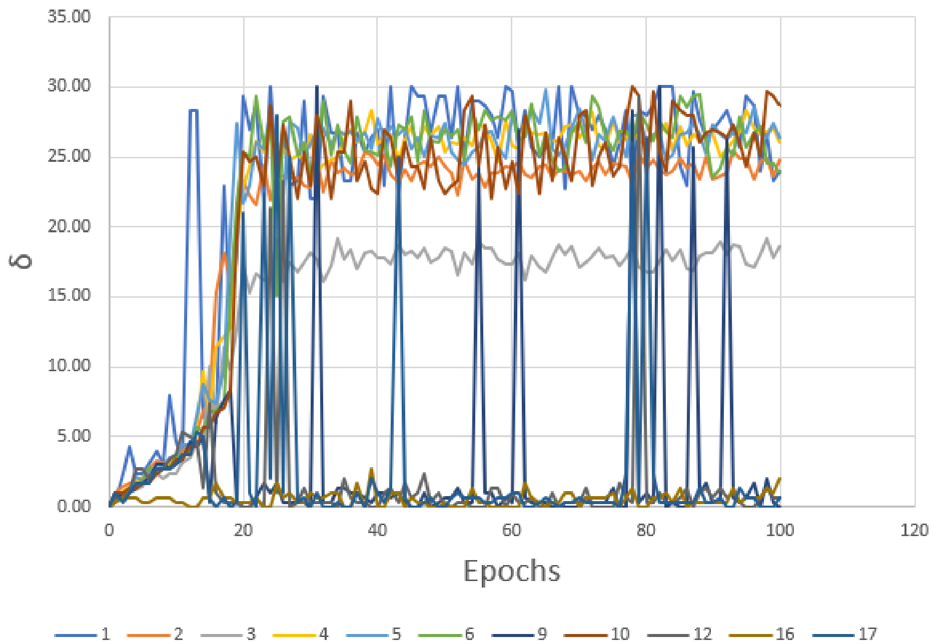

Figure 3.16: Mean δ by vertex degree in karate club network when $k = 10$ invitations per round

these networks with only one neighbor who does not need to compete for their invitation, the desire of these neighbors to attract other invitations keeps the singleton verteces from being taken advantage of. We therefore conjecture that this is why humans in social settings generally behave in a trustworthy and cooperative manner, even when they have the chance to take advantage of each other. It is only when the model is taken to extremes that we observe this behavior in social networks: Figure 3.16 illustrates this phenomenon in Zachary's karate club network when we let $k_i = 10$ for each agent i, with all other parameters the same as in Section 3.5. The figure displays the mean value of δ for all verteces of the same degree, and we see that while the verteces with fewer neighbors continue to compete for invitations, the verteces with more neighbors stop competing in order to take advantage of the invitations they receive, similar to the 5-star graph.

In addition to the numerical and theoretical results presented and discussed in this paper, there is still exploration to be done within the model. Empirically, the interactions we examined were strictly nonnegative between any two agents to avoid computational cost. We expect to see different behavior if this is changed so that the interactions are only nonnegative *in expectation*. This is a reasonable avenue of exploration, as sometimes partnerships may not work out despite positive expectations. Another interesting setting is non-uniform interactions between agents. For example, suppose that agent i provides a better partnership than agent j. How will δ_i and δ_j change in reaction to the utilities each can provide? We expect to see "diva"-like behavior in this case, with i displaying a low δ_i and still attracting many more partners than j.

3.7 Conclusion and Future Directions

In this chapter we considered pairwise interactions between agents arranged in a social network. We sought to determine how agents behaved when they had to compete with each other for interaction opportunities, using the limited-trust equilibrium to define player interactions. The agents in the network evolved to display behaviors which mirrored various empirical findings on human interaction, particularly [39] and [56] from the field of evolutionary biology. This is particularly notable as agents within this model were not forward thinking as real humans are: each focused on displaying a level of trustworthiness which was a best response to that currently being displayed by other players. Yet despite their lack of foresight or hindsight, the model motivates agents to behave in a trustworthy fashion without these considerations or historically-based mechanisms such as Grim Trigger or Tit-for-Tat.

In addition to empirical results of the model, a thorough mathematical analysis was presented. Simple learning algorithms were derived allowing agents to learn about their neighbors through interactions. A process for agents to update their trustworthiness metric δ based solely on their two-hop neighborhood within the network was also presented, and a Nash equilibrium was shown to exist in the δ-selection metagame that agents engage in. Along with the algorithms and processes which define how agents within the network evolve over time, mathematical bounds were also developed for the expected time for an agents to learn each others' δ to a given level of precision. These bounds can be viewed in Appendix B.1.

The model and experiments performed in this paper present a wide array of options for future research. While we noted that agents need not utilize mechanisms such as Grim Trigger or Tit-for-Tat in the social networks we considered, we also saw that limited-trust is not enough to encourage trustworthy behavior in the networks in Figures 3.14 and 3.15. Therefore, it will be interesting to incorporate these long standing concepts into the social network model proposed here. We believe that doing so will allow our model to more accurately represent interactions in smaller communities which run the risk of exhibiting the behaviors discussed, such as diades.

We are also interested in incorporating historical data more generally into the decision making process. Individuals who interact frequently are more likely to have an established relationship. They are thus less likely to switch partners if the utility increase is minor. Another way to incorporate historical data is based on past utility earned. Agents who do well are able to increase the number of interactions they can initiate per round. Changes to network structure based on past behavior is a related topic: edges may wither if unused, or new edges may appear from an agent i to an agent j if the two have a mutual neighbor l whom both interact with frequently.

One final area of interest is how agents behave with unknown network structure. Consider the case in which an agent i is only aware of its one-hop neighborhood N_1^i, and cannot know for certain its two-hop neighborhood when updating δ_i. We would like to develop methods for i to estimate the structure of N_2^i as doing so will remove the need for an agent i to know δ_j' of its neighbor j.

It is our hope that the wider research community is as excited by this work as we are. In order to make it easier for interested researchers to explore this setting, we have made a portion of our code publicly available on Github at `https://github.com/kzr-soze/SocialNetworkGames`. This repository is written in Python3 and contains the class file which can be used to generate a network and track player behavior, as well as demo scripts to show its use. We hope that making it available will enable further exploration by interested researchers.

CHAPTER 4

PRIZE-COLLECTING MULTI-AGENT ORIENTEERING: LIMITING INEFFICIENCY DUE TO SELFISHNESS

This chapter is based on work which has been accepted for publication in *IEEE Transactions on Automation Science and Engineering*.

4.1 Introduction

Previously in this dissertation, we considered agents who did not behave in a completely selfish manner. In this chapter, we return to the traditional assumption of selfishness and instead focus on policies to mitigate the effects of this selfishness from the perspective of a game's organizer.

Team Orienteering Problems (TOPs) began as outdoor games: players arrive in the woods and are equipped with compasses, maps, and instructions for finding checkpoints, and must visit as many checkpoints as possible within a given time limit. These games have a natural link to the classical Vehicle Routing Problem (VRP), but are distinct from the VRP in that it may not be possible to visit all checkpoints; the players must decide where they should go in order to collect the maximum number of points given that not all checkpoints are equal. The TOP is NP-hard even in the single agent case; see [59].

The TOP arises naturally in logistics as an extension of the VRP: direct-to-customer shipping companies must decide which of their orders should be filled today rather than tomorrow when it is infeasible to fill all orders, and retail companies must determine which outlets need to be resupplied immediately to maintain positive inventory and which can wait until next week. Fittingly for such an important problem, it has been well-studied and heuristics have been proposed with empirically satisfying results (more on this in Section 4.1.1).

The situation which we were interested in modeling as a TOP is an Unmanned Aerial Vehicle (UAV) or drone Intelligence Surveillance and Reconnaissance (ISR) network. However, we found that the traditional TOP setup could not realistically model this scenario: TOP problems assume centralized solutions or communication between teammates, as all are working toward a common goal. This is not a realistic assumption for a drone intelligence-gathering network, particularly in an adversarial setting where incoming and outgoing communications may be observed or denied by jamming, and team members may be out of communication with the central authority for extended periods of time. Agents may be aware of their teammates' locations through passive sensing techniques such as visual detection but are unable or unwilling to communicate more actively by broadcasting information. Therefore, we consider a natural approach to this problem, allowing each team member to act as an independent agent seeking to maximize its own score. By reasoning about their fellow agents' locations, agents attempting to maximize their own scores will naturally try to minimize their overlap. We designate this setting as the Prize-Collecting Multi-Agent Orienteering Problem (PCMOP), a new variation on the TOP. The PCMOP is distinct from the Multiagent Orienteering Problem (MOP) formulated by [30] as prizes can be collected by only one agent, with rules regarding which agent may collect them determined by a policy over the game. More detail on these distinctions will be given in Section 4.1.1. We propose three such policies to determine how prizes can be distributed among the agents, and examine the resulting total prize collection. We consider the equilibria resulting from our policies on different graphs such as general, undirected, and directed acyclic, and calculate the theoretical Price of Anarchy (PoA) related to them as a measure of the maximum inefficiency of these equilibria compared to a centrally coordinated optimal solution. The goal of the fleet operator is to have agents collect a maximum value set of prizes through the selection of an appropriate policy.

The rest of the paper is organized as follows: the following subsection comprises an in-depth literature review of work related to the traditional TOP, as well as the few papers which address situations with self-interested agents. Section 4.2 details the setting of our problem, and provides a full description of the three policies we propose and examine. In §4.3, we analyze our policies for different network types for 2 agents and develop tight bounds on the PoA for each policy, as well as an extension to the result on the

PoA of simultaneous games over *valid utility systems* [2], showing that they display a PoA of at most 2. In §4.4, we repeat the analysis for an arbitrary number of players k. In §4.5, we develop methods for solving the Stackelberg games resulting from each of our proposed prize-division policies. In §4.6, we numerically analyze on test cases by generating approximate $\mathcal{R}^2$ and $\mathcal{R}^2$ planar networks. In §4.7, we present a summary and discussion of our results. A full summary of our theoretical results is given in Table 4.2 organized by network type, policy type, number of players, range of players, and whether players represent a homogeneous fleet (the details of the policies are defined in §4.2).

4.1.1 Literature Review

As mentioned in the introduction, the TOP is well-studied and several heuristic and exact solution approaches have been proposed and empirically tested in papers such as [59–66]. [67] performs an in-depth analysis of the single-agent case. [68] examines the applicability of the TOP to drone-related military situations and [69] addresses a UAV variant in which each prize is information, and depending on UAV configuration a single agent may only collect certain types of information from each location. Additionally, [70] addresses the problem of managing a UAV fleet in a communication denied area where time to complete actions and the reward for doing so are uncertain.

Many works, such as [71–78], propose frameworks and algorithms for coordination of UAV fleets in different settings, such as combating wildfires, surveillance, and target removal, and do so under both online and *a priori* knowledge settings.

Another work somewhat closely related to ours is [30], which formulates and addresses the MOP, and relies on a game-theoretic framework for analysis. In particular, a node i with a prize p_i takes time t_i to deliver that prize to any player who visits it. The node i can only service k_i players at a time, leading to a queue if more than k_i players are present. However, while the number of players who may receive a prize at once on a node is a limiting condition, once an earlier player leaves, the next queued player can still receive a prize. This game models theme-park or tourist routing situations, in which players want to visit the most attractions with the fewest lines, but does not adapt to the situation of limited prizes which we consider, necessitating our PCMOP model. However, the work from which this paper most draws inspiration is the recent article [79], which considers the problem of area surveillance with self-interested agents. In it, two or more UAVs greedily consider choosing routes for the next l time-steps. Additionally, different types of information/prizes are available at each site, which can only be collected by a specific UAV if it has the appropriate sensor type (audio, video, thermal, etc.). The sensors have varying levels of effectiveness so that some portion of the collectible information type is captured and the residual information of that type may be collected by another UAV. However, there are two fundamental differences between [79] and our work: [79] considers simultaneous movements in the game, while our paper considers variations on leader-follower strategies. We will see in Sections 4.3 and 4.4 that the leader-follower setting introduces several new complications. Additionally, [79] focuses on games over spacial grid graphs, while we consider games over more general networks.

The proposed PCMOP model has the following similarities with congestion games [80]: both consist of multiple self-interested agents attempting to get from source to destination. However, congestion games focus on the edges of the graph as the source of the delay, while the PCMOP focuses on the nodes of the graph as the locations of prizes. Further, unlike in a congestion game where each player of the same type traveling along an edge experiences the same delay, variations on the prize collecting problems provide inherently unequal payoffs as prizes can only be collected by one individual. However, the two classes of games are similar enough that we look to similar tools in order to analyze them: [81,82] and [83] all describe variations on Stackelberg (leader-follower) strategies for player decision-making in the congestion game which we consider in our analysis of the PCMOP. In this paper, we focus on a Stackelberg setting in which a pure equilibrium is guaranteed to exist, as opposed to the simultaneous setting (meaning that both players choose their routes simultaneously rather than in leader-follower ordering) in which a PNE does not necessarily exist; see Lemma 4. However, a leader-follower setting also introduces complications to decision-making and analysis, which we detail in Sections 4.3 and 4.4.

4.2 Setting

A Prize-Collecting Multi-Agent Orienteering Problem is defined by a graph $G(V, E, R)$ and agents $P(D, \mathcal{S}, \mathcal{T})$ where $|P(D, \mathcal{S}, \mathcal{T})| = k$ is the number of agents. Here,

- $V :=$ the set of vertices/nodes in the network, $|V| = n$.
- $E :=$ the set of directed or undirected edges in the graph. Edge e has a nonnegative length l_e.
- $R :=$ the set of prizes at each vertex.
- $D :=$ the set of maximum distances each player can travel before they must reach their destination.
- $\mathcal{S} :=$ the set of source nodes for each player, i.e. where they begin their route.
- $\mathcal{T} :=$ the set of terminal nodes for each player, i.e. where they must end their route.
- $\Sigma :=$ the set of strategy spaces of each player, with Σ_i the set of mixed strategies of player i; a mixed strategy is a probability distribution over the pure strategies, which in this context are the set of all paths from s_i to t_i.

We consider positive edge lengths and non-negative prizes as we are concerned primarily with the drone surveillance network use-case, and a graph network is a convenient abstraction from the $\mathcal{R}^2$ setting that prizes (areas of interest for surveillance) are likely to be in. Graphs can be constructed as a grid network, or solely as edges between the prize locations (nodes) depending on operator preference.

We consider the general case of this problem, where player sources, terminals, and maximum distances may vary, although traversal speed is assumed to be equal for all players. However, special consideration will also be given to the case of a homogeneous fleet, in which each player has the same maximum distance d, a common source S, and a common terminal T. We will consider the homogeneous case specifically due to its relation to our underlying use case: operators of a fleet of surveillance drones will likely have a homogeneous fleet, and may well be operating from the same origin/destination ($s_i = s_j, t_i = t_j \ \forall i, j$). A natural question for a manager or fleet commander in this setting is how to deploy its agents such that the inefficiency due to agent selfishness is minimized. That is, the manager wants its agents to collect a set of prizes so that the net value of prizes collected by all agents is maximized. This is in contrast to the agents, who want to collect as set of prizes of maximum value for themselves. We will consider three potential deployment policies.

4.2.1 Policies to explore

We propose and explore the efficiencies of three natural policies. We assume an arbitrary ordering of the players $1, 2, ..., k$ to indicate turn-ordering in a leader-follower setting. We believe both the policies and the ordering to be natural to large organizations, which frequently display hierarchies based on seniority (e.g., nurses or flight attendants picking schedules). The policies are:

1. Reserved path policy: A player i declares its path from s_i to t_i. All prizes it claims cannot be picked up by another player $j > i$, even if j arrives first.

2. Unreserved path policy (priority for ties): A player i declares its path from s_i to t_i, and may not deviate from it. If another player j with lesser priority (i.e., $j > i$ in the ordering) arrives at a node first, j collects the prize. If both i and j arrive simultaneously, i collects the prize in its entirety and j receives nothing.

3. Turn-based policy: Players take turns moving through the network one node at a time. This corresponds to players moving simultaneously through the network but only having to commit to the next node, rather than their whole route. In the event that two players i and $j > i$ arrive at a node simultaneously, i collects the prize in its entirety and j receives nothing.

We use the term "full-path" policies to refer to the reserved and unreserved path policies, as each player i's strategy set consists of full paths from

s_i to t_i. Additionally, we assume that agents are not capable of waiting in a single location, i.e. choosing not to move, as areas which need constant surveillance will have permanent cameras installed. However, waiting can be easily incorporated by adding a self-loop on each node in the network. Waiting will only occur in the turn-based policy though, as it is sub-optimal in the reserved and unreserved path policies. We choose these policies because, due to the successive manner in which players choose their moves, each can be considered an instance of a Stackelberg game in which players take turns selecting their strategies, and the actions of earlier players can be observed by later players. This guarantees the existence of a pure equilibrium. While Stackelberg defined the equilibrium for games of 2-players, the concept can be generalized to k players. More formally:

Definition 10. *A k-player leader-follower (Stackelberg) game with a leader-follower ordering of players $\{1, 2, ..., k\}$ is said to display a pure Stackleberg equilibrium when no deviation by player i will result in a higher payoff for player i, taking into account the changes that players $i + 1$ through k will make to their strategies in response.*

We adopt this k-player extension of the Stackelberg equilibrium from [84].

Lemma 3. *Any k-player full-knowledge leader-follower game displays a pure equilibrium provided the maximum total number of strategy decisions for the game is finite and players have a fixed rule for breaking ties between strategies with equivalent payoffs.*

Proof. Consider player k, the last player to move. Player k must pick the strategy which benefits it the most with full knowledge of the strategies chosen by the first $k - 1$ players. Therefore, the choice which maximizes its payoff given those strategies is a pure equilibrium choice. In the event of an equal maximal payoff between multiple strategies, player k picks according to the fixed tie-breaking rule. Player $k - 1$ can predict exactly how player k will react to its own strategy and has full knowledge of the first $k-2$ players. The use of a fixed-rule for choosing between ties ensures this. Therefore, the choice which maximizes its payoff given those strategies and player k's response is a pure equilibrium choice. Similarly, player i knows the strategy choices of the first $i - 1$ players, and can predict how player $i + 1$, and by extension all players after it, will react to its own strategy. Thus, picking the strategy which maximizes i's payoff given the chosen strategies and coming (predictable) reactions is a pure equilibrium for i. Therefore, under this policy, every player has a pure equilibrium choice regardless of what the previous players did. $\square$

We note that we require the maximum total number of choices made to be finite, not the total number of choices available: On an undirected graph, an agent with infinite range has an infinite number of routes, as it may cycle indefinitely. However, it has only a finite, albeit exponentially large in $|E|$, number of routes it should consider as cycling will not result in any increase in payoff. We note that when each agent i has finite range $d_i \in D$, this is not a problem as there are only a finite number of choices available. With non-finite d_i we resolve this issue by assuming that given two strategies with equal payoff from the same set of nodes, an agent will pick the one corresponding to a shorter route (i.e. avoiding needless cycling). This removes the possibility of non-terminating routes in the reserved and unreserved path policies. However, it does become a problem in the turn-based policy when agents have unrestricted range, as it may result in non-terminating routes. Theorem 13 shows this in more detail. We also note that having an agent with non-finite range is not feasible in our motivating use case, but we believe it important to consider how agents behave in extreme settings.

While the term PoA traditionally refers to the ratio of the optimal centrally coordinated solution to the worst Nash equilibrium when discussing utility maximization games, we will use it here to refer to the ratio of the optimal centrally coordinated solution to the worst Stackelberg equilibrium where PoA ≥ 1. More formally, for a game g, $PoA(g) = \max_{\sigma \in SE(g)} \frac{u(\sigma^*)}{u(\sigma)}$ where σ^* is the centrally coordinated solution. It is also standard practice to denote the PoA of a set of games G as the supremum of the PoAs of the games in the set, $PoA(G) = \sup_{g \in G} PoA(g)$. Finally, we will be interested in two specific fixed tie-breaking rules in this paper, *Goodwill* and *Sadism*. The rule of Goodwill will limit or reduce some of the performance inefficiencies of our three policies, and the corresponding rule of Sadism will increase these inefficiencies.

Definition 11. *A player i is said to display goodwill to a player j if, given a set of strategies Σ_i all resulting in equal (maximal) payoff for i, player i picks the one which allows j to achieve the maximum payoff.*

Definition 12. *A player i is said to display sadism to a player j if, given a set of strategies Σ_i all resulting in equal (maximal) payoff for i, player i picks the one which forces j to achieve the minimum payoff.*

Under any fixed rule which fails to break a tie, we assume that the player choosing makes its choice according to some second arbitrary rule, such as a lexicographic ordering of routes, which will not fail.

Another policy to consider would be one of simultaneous route picking, in which each agent simultaneously picks its entire route and proceeds through the network, collecting any prizes it comes across first. However, we have largely neglected to explore this policy because it does not necessarily possess a PNE as shown in the following lemma.

Lemma 4. *Consider a 2-player game in which both players simultaneously pick their entire routes and split any prizes they arrive at simultaneously according to some fixed proportion $\lambda \in [0,1]$. This game does not necessarily contain a PNE.*

Proof. We show this by a counter-example. Consider the network in Figure 4.1. There are 3 routes between S and T: ABC, AC, and D. The value underneath each node label is the prize associated with that node, i.e. node C has a prize of $2 - 4\varepsilon$, where $0 < \varepsilon \ll 1$. Players are identical: both start from S and go to T, with $d_1 = d_2 \geq 4$ and $l_e = 1$ for all edges $e \in E$. If both players arrive simultaneously at a node, player 1 receives $\lambda \in [0,1]$ of the prize and player 2 receives $\mu = (1 - \lambda)$. Due to the small number of routes, we construct the payoff matrix for both players picking their entire route simultaneously in Table 4.1, where player 1 is the row player and player 2 is the column player. There is no value $\lambda \in [0,1]$ in which causes a cell in the matrix to be a PNE. □

Table 4.1: Payoff matrix for a full-route simultaneous game on Figure 4.1

Route	ABC	AC	D
ABC	$\lambda(3-2\varepsilon)$, $\mu(3-2\varepsilon)$	$1+(1+\lambda)\varepsilon$, $2-(3+\lambda)\varepsilon$	$3-2\varepsilon$, $1+2\varepsilon$
AC	$2-(4-\lambda)\varepsilon$, $(2-\lambda)\varepsilon$	$\lambda(2-3\varepsilon)$, $\mu(2-3\varepsilon)$	$2-3\varepsilon$, $1+2\varepsilon$
D	$1+2\varepsilon$, $3-2\varepsilon$	$1+2\varepsilon$, $2-3\varepsilon$	$\lambda(1+2\varepsilon)$, $\mu(1+2\varepsilon)$

We conclude this section with a compilation of our theoretical results, presented in Table 4.2. We use $\mathfrak{e}$ to represent Euler's number, ≈ 2.718, as e is already used for edge representation.

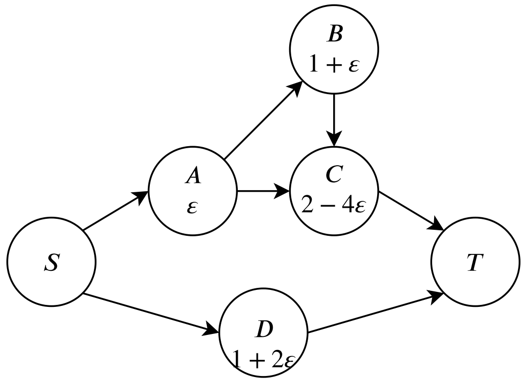

Figure 4.1: Network with no pure Nash equilibrium for a 2-Player full-route simultaneous game

4.3 Results: The 2-Player PCMOP

We now provide the proofs of the results for 2-player games.

4.3.1 The General 2-Player PCMOP

Before providing results related to the PoAs of the reserved and unreserved path policies, we first revisit the idea of simultaneous games under these full-path policies. Lemma 4 shows that a PNE may not exist in all games for this setting. Therefore, we now derive PoA bounds under mixed Nash equilibrium, which is guaranteed to exist [1], and we show that the PoA under the reserved and unreserved path policies is at most 2. We will do so by drawing upon the concept of a *valid utility system* from [2].

A *utility system* is defined with the following structure: Non-cooperative agents whose action spaces are subsets of an underlying groundset make decisions which induce some social utility, measured by a set function on the actions taken. The agents attempt to maximize their own private utility rather than the social utility. Additionally, the following three conditions hold:

1. The social utility function u and the private utility functions u_i are measured in the same standard unit.

2. The social utility set-function u is submodular. Mathematically, for $A \subseteq B$ and $x \notin B$, we have $u(A \cup \{x\}) - u(A) \geq u(B \cup \{x\}) - u(B)$.

Table 4.2: PoA Bounds for studied policies on different network types

		Homogeneous Fleet		General Fleet	
Network	Policy	2-Player	k-Player	2-Player	k-Player
Directed Acyclic	Reserved Path	$\frac{4}{3}^*$	$\frac{k^k}{k^k-(k-1)^k} \to \frac{\mathfrak{e}}{\mathfrak{e}-1}^*$	2^*	2^*
	Unreserved Path	2^*	$\frac{k^2}{k-1}$	2^*	$\frac{k^2}{k-1}$
	Turn-Based	Unbounded	Unbounded	Unbounded	Unbounded
General	Reserved Path	$\frac{4}{3}^*$	$\frac{k^k}{k^k-(k-1)^k} \to \frac{\mathfrak{e}}{\mathfrak{e}-1}^*$	2^*	2^*
	Unreserved Path	2^*	$\frac{k^2}{k-1}$	2^*	$\frac{k^2}{k-1}$
	Turn-Based	Unbounded†	Unbounded†	Unbounded*	Unbounded*
Undirected	Reserved Path	$\frac{4}{3}^*$	$\frac{k^k}{k^k-(k-1)^k} \to \frac{\mathfrak{e}}{\mathfrak{e}-1}^*$	2^*	2^*
	Unreserved Path	2^*	$\frac{k^2}{k-1}$	2^*	$\frac{k^2}{k-1}$
	Turn-Based	Unbounded	Unbounded	Unbounded	Unbounded
Undirected (Unrestricted Range)	Reserved Path	1^*	1^*	1^*	1^*
	Unreserved Path	1^*	1^*	1^*	1^*
	Turn-Based	1^*	1^*	1^*	1^*

* Bound is tight
† Non-Terminating

3. The private utility of an agent i is at least the change in social utility which would occur if the agent did not participate in the game. For a strategy set σ and σ_{-i}, the actions of all other agents, we have that $u_i(\sigma) \geq u(\sigma) - u(\sigma_{-i})$.

A utility system is *valid* if and only if

4. $\sum_{i=1}^{k} u_i(\sigma) \leq u(\sigma)$ for all strategy profiles σ.

[2] shows that any game over a valid utility system has a PoA of at most 2.

Theorem 9. *Any simultaneous game under the reserved or unreserved path policies has a PoA of at most 2.*

Proof. This proof will proceed by showing that in the simultaneous setting, the game under the reserved and unreserved path policies is a game over a valid utility system, as defined in [2]. This is sufficient, as [2] also shows that any k-player game over a valid utility system has a PoA of at most 2.

1. The social utility function and player utility functions are both measured in the same units: the value of the prizes collected.

2. The social utility function is submodular. To see this, suppose we have two sets of player paths, S and S' such that $S \subseteq S'$. For some path p, we have that $u(S \cup \{p\}) - u(S) \geq u(S' \cup \{p\}) - u(S')$ as the set of prizes on p which are uncollected in S' must be a subset of the set of prizes on p which are uncollected in S.

3. Private utility of each player is at least as much as the change in the social utility if that player was not present and all other players played the same strategy: The change in the social utility from player i being present is exactly the value of the prizes which are on its path p_i and not any of the other paths p_{-i}, and player i receives at least this set of prizes under both policies.

4. The sum of the players' private utilities is at most the value of the social utility function. This is equivalent to saying $\sum_{i=1}^{k} u_i(S) \leq u(S)$ for any set of paths S. Here the social utility is defined to be the value of all prizes obtained which means that $\sum_{i=1}^{k} u_i(S) = u(S)$.

This completes our proof. □

Next, we provide an extension of [2]'s proof to show that any 2-player Stackelberg equilibrium over a valid utility system also has a PoA of at most 2 when the social and private utility functions are $u_i(S_{-i} \cup \varnothing_i) = 0$ for all $S_{-i} \in \Sigma_{-i}$ for all players i. Here $\varnothing_i$ is equivalent to player i taking no action.

Theorem 10. *Given a 2-player leader-follower game over a valid utility system in which $\sum_{i=1}^{2} u_i(S) = u(S)$ and $u_i(\varnothing_i) = 0$, the PoA is at most 2.*

Proof. This will be proven by constructing a new simultaneous game in which there is a PNE equivalent to the leader-follower equilibrium, then showing that the setting over which the new game is played is still a valid utility system. Let $S^{st} = \{s_1^{st}, s_2^{st}\}$ be the Stackelberg equilibrium. Define $BR_2(s_1) = \arg\max_{s_2 \in \Sigma_2} u_2(s_1, s_2)$ as the second player's best response to the first player playing s_1. We next introduce a new pure strategy $*_2$ for player 2, where $u_1(s_1, *_2) = u_1(s_1, BR_2(s_1))$ and $u_2(s_1, *_2) = u_2(s_1, BR_2(s_1))$ for all $s_1 \in \Sigma_1$. Thus playing $*_2$ is equivalent to player 2 playing its best response to s_1 after observing s_1. Because $*_2$ is the best response to every pure strategy s_1, it is also the best response to every mixed strategy σ_1. $*_2$ is thus a (possibly weakly) dominant strategy for player 2, and therefore there exists at least one PNE $\{s_1^{st}, *_2\}$. This equilibrium is equivalent to the Stackelberg equilibrium S^{st}. As an example, consider playing the simultaneous prize-collecting game over the network in Figure 4.1 under the unreserved policy: After the introduction of $*_2$ there is a pure equilibrium of $(AC, *_2)$, which is equivalent to the Stackelberg equilibrium of (AC, D).

Next, we show that the new game still represents a game over a valid utility system as defined by [2]. As the original game was over a valid utility system, we only need to consider what happens when the second player plays $*_2$. However, we first note that if player i takes action s_i and player $-i$ takes no action, then $u(s_i) = u(s_i, \varnothing_{-i}) = u_i(s_i, \varnothing_{-i}) + u_{-i}(s_i, \varnothing_{-i}) = u_i(s_i, \varnothing_{-i}) = u_i(s_i)$.

1. The social utility and players' personal utilities are still measured in the same units. This is because the original game was over a valid utility system, and the utility functions have not changed with the introduction of $*_2$.

2. The private utility each player receives is at least as much as the change in social utility from their action. To demonstrate this, we must show $u_2(s_1, *_2) \geq u(s_1, *_2) - u(s_1)$ and $u_1(s_1, *_2) \geq u(s_1, *_2) - u(*_2)$ where s_1 is any action taken by player 1. $u_2(s_1, *_2) \geq u(s_1, *_2) - u(s_1)$ follows from the fact that the original game was over a valid utility system and playing $*_2$ is equivalent to playing $BR_2(s_1)$, player 2's best response to s_1. For the second, let $s_2 = BR_2(s_1)$ be the second player's best response to s_1 and let $s_2' = BR_2(\varnothing_1)$ be player 2's best action when under no competition. Clearly $u(s_2') = u_2(s_2') \geq u_2(s_2) = u(s_2)$. Therefore,

$$\begin{aligned} u_1(s_1, *_2) = u_1(s_1, s_2) &\geq u(s_1, s_2) - u(s_2) \\ &\geq u(s_1, s_2) - u(s_2') \\ &= u(s_1, *_2) - u(*_2). \end{aligned}$$

3. The social utility function is submodular. To show this, we note there are only two players and show that $u(*_2) - u(\varnothing) \geq u(s_1, *_2) - u(s_1)$ and

$u(s_1) - u(\varnothing) \geq u(s_1, *_2) - u(*_2)$. The first follows from the fact that the original game was over a valid utility system: adding $*_2$ to the set is equivalent to adding $s_2 = BR_2(s_1)$, player 2's best response to s_1, on the right hand side. On the left hand side, it is equivalent to adding $s_2' = BR_2(\varnothing_1)$, player 2's best action when under no competition. Therefore

$$\begin{aligned} u(*_2) - u(\varnothing) = u(s_2') &= u_2(s_2') \\ &\geq u_2(s_2) \\ &= u(s_2) \\ &\geq u(s_1, s_2) - u(s_1) \\ &= u(s_1, *_2) - u(s_1), \end{aligned}$$

where $u_2(s_2') \geq u_2(s_2)$ was established in the previous point. For the second, from the original game we know that $u(s_1) \geq u(s_1, s_2) - u(s_2)$. We also know $u(s_2) \leq u(s_s')$ which implies

$$\begin{aligned} u(s_1) - u(\varnothing) &\geq u(s_1, s_2) - u(s_2) \\ &\geq u(s_1, s_2) - u(s_2') \\ &= u(s_1, *_2) - u(*_2). \end{aligned}$$

4. $\sum_{i=1}^{2} u_i(S) \leq u(S)$. As we have assumed $\sum_{i=1}^{2} u_i(S) = u(S)$, and the utility functions have not changed with the introduction of $*_2$, this is true.

This completes the proof. □

It is immediately apparent that Theorems 9 and 10 together imply an upper bound of 2 on the PoA of 2-player games under the unreserved path policy. In Theorem 12 we will show that this bound is tight.

Lemma 5. *The PoA in the general k-Player setting under the reserved path policy is at most 2.*

Proof. Theorem 9 demonstrates that in the simultaneous setting, the k-Player game under the reserved path policy represents a game over a valid utility system. Therefore, [2] implies that under the simultaneous setting, the game has a PoA of at most 2. We make the observation that under the reserved path policy, the leader-follower and simultaneous games are equivalent as for $i < j < l$, player j can ignore the actions of player l and predict

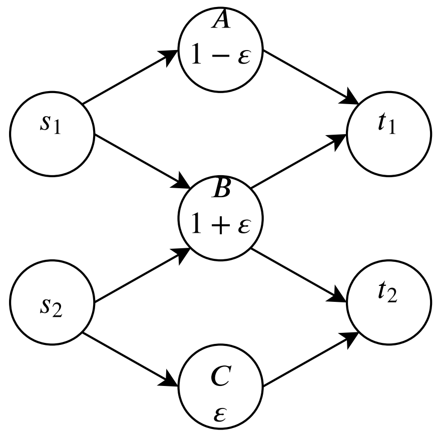

Figure 4.2: Network With PoA of 2 under reserved and unreserved path policies

the actions of player i, as i can also ignore all players with less "seniority" than it has, even in the simultaneous setting. Therefore, the reserved path policy has a PoA of at most 2 in the general setting, as it is equivalent to a simultaneous game over a valid utility system. Figure 4.2 shows this bound to be tight using $k = 2$ players: The first player will go to B and the second will go to C before continuing to t_1 and t_2, respectively. A total of $1 + 2\varepsilon$ in prizes will be collected, when the centrally coordinated solution would collect a total of 2 in prizes, from nodes A and B. □

4.3.2 The Homogeneous 2-Player PCMOP

Previously, we considered the most general form of the 2-Player PCMOP. Now we consider the homogeneous fleet PCMOP, where $s_i = s_j = S$, $t_i = t_j = T$, and $d_i = d_j = d$. We refer to such a setting as a *homogeneous game* and we will see that although the PoA for the 2-player game under the unreserved path policy remains 2, the PoA for the reserved path policy will be reduced to $\frac{4}{3}$.

In order to prove several of our results in this section, we first let $A_{[i]}$ denote the total value of the prizes that are on the routes planned by players 1 through i. We let $A^*_{[k]}$ denote the total value of the prizes collected by k players in the optimal centrally coordinated solution. We will typically normalize $A^*_{[k]} = 1$ when proving theorems.

Lemma 6. *In a k-player homogeneous game in which players 1 through i have planned their routes to collect a total of $A_{[i]}$ prizes, there is a path containing at least $\frac{1}{k}(A^*_{[k]} - A_{[i]})$ prizes which none of the first i players will collect in the unreserved and reserved path policies.*

Proof. If the first i players have set their routes so that their paths contain a set of prizes valued at $A_{[i]}$ in total, then the optimal paths of the k players must still retain at least $(A^*_{[k]} - A_{[i]})$ in ignored prizes. There are k optimal paths, so at least one must contain a set of prizes with minimum value of $\frac{1}{k}(A^*_{[k]} - A_{[i]})$ which is non-overlapping with the set of prizes in the first i players' paths. □

Theorem 11. *Under the reserved path Policy, the PoA of 2-player homogeneous games has a tight upper bound of $\frac{4}{3}$.*

Proof. Lemma 6 directly provides an upper bound of $\frac{4}{3}$ on the PoA: If we normalize $A^*_{[2]} = 1$, the value of the prizes collected by two players in the centrally coordinated solution, there is a path containing $a_1 \geq \frac{1}{2}$ in prizes which the first player collects. With that in mind, the Lemma then shows there is a path containing $a_2 \geq \frac{1}{2}(1 - a_1)$ in uncollected prizes which is taken by the second player, as it cannot steal any prizes from the first player. The total collected is $a_1 + a_2 \geq \frac{3}{4}$. We show this bound is tight by example, using the network in Figure 4.3. The network displays a PoA of $\frac{4}{3}$, so this is a tight bound for the reserved path policy in a directed acyclic, and therefore general, graph: In the figure, the first player will maximize its payoff by going to the two nodes containing prizes of $1 + \varepsilon$, leaving the second player able to collect only one of the remaining prizes. We also show it to be tight on an undirected graph with restricted range d using the same example. We do so by setting the range to $d = 3$ and changing the edges in Figure 4.3 to be undirected. Player 1 again collects the two $1 + \varepsilon$ prizes and Player 2 again collects only one of the two remaining prizes. Therefore, the PoA is $\frac{4}{3}$, so this is a tight bound for undirected graphs with limited range as well. □

Theorem 12. *Under the unreserved path Policy, the PoA of 2-player homogeneous games has a tight upper bound of* 2.

Proof. Theorems 9 and 10 together imply that in the general 2-player game the unreserved path policy has a PoA of at most 2 for all networks. Now we

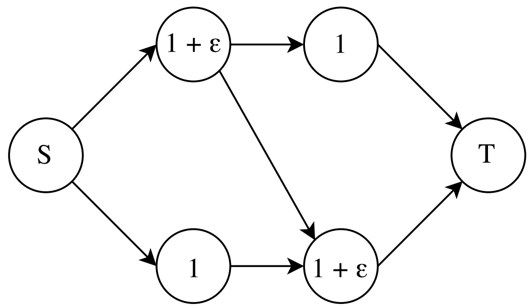

Figure 4.3: Network with a 2-Player PoA of $\frac{4}{3}$ under the reserved path Policy

show by example that this bound is tight for identical 2-player games: Figure 4.4 demonstrates a PoA of 2 on a directed acyclic graph (DAG) and hence a general graph, with the first player choosing $B \to C \to T$ and the second choosing $C \to T$. If the network in the figure is undirected and each player has a maximum range of 4, then the first player again chooses $B \to C \to T$ while the second player now chooses $C \to T \to E \to T$. Thus the bound of 2 also applies to undirected networks with limited range. □

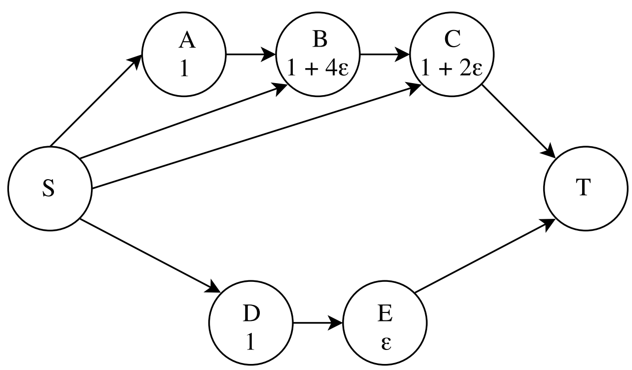

Figure 4.4: Network with a 2-Player PoA of 2 for Turn-Based and Unreserved Policies

Thus far we have not addressed the turn-based policy. This is because even in the 2-player setting it must be considered as an extensive-form game, something which we can avoid in the reserved and unreserved path policies. Therefore, most of our work with the turn-based path policy is presented in Sections 4.5 and 4.6, as an empirical study. However, we will at this time provide one theoretical result:

Theorem 13. *Under the turn-based policy, the Price of Anarchy for a k-player homogeneous game may be unbounded for an arbitrary fixed tie-breaking rule.*

Proof. This will be a proof by example using a game with $k = 2$. Suppose there is a bound r on the PoA of turn-based games in general graphs. We will construct a game which has a PoA greater than this. Consider the graph in Figure 4.5 and a 2-player game where each player has range $d = r + 3$ and the directed ring of 1-prize nodes is of length greater than $2(r + 1)$. There are directed edges from both A and B to every node in the ring, and every ring node also has a directed edge going to T. Centrally coordinated, each player should move to one of the staging nodes (A and B) and then move to the ring in such a way that they can each collect $r + 1 + \varepsilon$ prizes before moving to T, resulting in $2(r + 1 + \varepsilon)$ prizes collected in total.

Now we consider the game when each player displays *sadism* toward the other. After both players' initial moves, each will be at A or B. Without loss of generality, assume player 1 is at A and player 2 is at B. The first player must decide whether to go to B or go to one of the nodes on the ring. If it goes to one of the nodes on the ring, the first player will collect a prize of one and the second player, being sadistic, will move directly in front of it resulting in the first player obtaining $1 + \varepsilon$ prizes. The first player is able to do this at any point, as the second player cannot collect all the ring prizes, so it obtains the same value in prizes by waiting. Additionally, it knows that the second player will not venture into the ring and so will reduce the number of prizes it can collect. Because the first player is sadistic, it therefore moves to B. The second player is then faced with the same choice and, as it is also sadistic, it moves from B to A. The two players cycle back and forth until each has two moves left, then each will visit one ring node before proceeding immediately to T. The value of the prizes collected is $2(1 + \varepsilon)$, giving a PoA of $\frac{2(r+1+\varepsilon)}{2(1+\varepsilon)} > r$. ∎

Note that Theorem 13 only implies that an arbitrarily chosen fixed tie-breaking rule may have an unbounded PoA, not that every fixed tie-breaking rule has one.

Note that as this result applies to homogeneous games, it also applies to the more general PCMOP setting. It is also worth noting that in the setting described in the proof of Theorem 13, if both players have infinite range they may cycle indefinitely, and the game will not terminate.

Despite the lack of formal theoretical bounds on the performance of the turn-based policy, we show in Section 4.6 that empirically it results in a

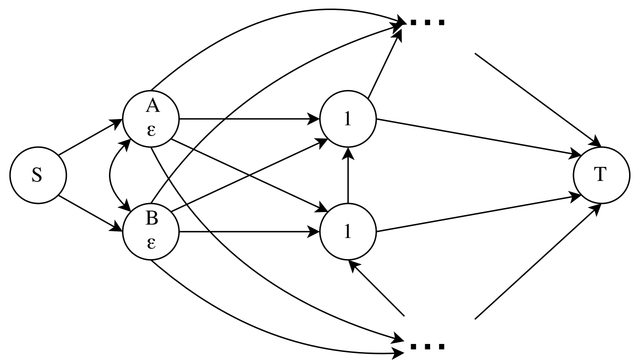

Figure 4.5: Network with an unbounded 2-Player PoA for Turn-Based Policy

lower average PoA across nearly all tested problem classes and sizes than the unreserved policy.

4.4 The k-Player PCMOP Game

We now wish to consider the PCMOP with an arbitrary number of players k. For the k-player game, we confine our discussion to the homogeneous setting.

From Lemma 5, we know the k-player reserved path policy has a PoA of at most 2, and that bound is tight. However, Theorem 11 shows the bound improves to $\frac{4}{3}$ in 2-player homogeneous games, leading to the following Theorem:

Theorem 14. *Under the reserved path Policy, the Price of Anarchy for homogeneous k-player games has a tight upper bound of $\frac{k^k}{k^k-(k-1)^k}$, with a limit of $\frac{\mathfrak{e}}{\mathfrak{e}-1}$ as $k \to \infty$.*

Proof. Suppose $A^*_{[k]} = 1$. Using Lemma 6 we know that the first player captures prizes with a value of $a_1 \geq \frac{1}{k}$. The second player captures prizes with a minimum value of $a_2 \geq \frac{1}{k}(1 - a_1)$ after player 1 plays, so players 1 and 2 together capture prizes with a minimum value of $\frac{2k-1}{k^2}$. Table 4.3 shows the results of continuing this line of reasoning. We next show that a lower bound on the value of the prizes captured in a k player game is $\frac{\sum_{j=0}^{k-1} \binom{k}{j} * k^j (-1)^{k-1-j}}{k^k} = \frac{k^k-(k-1)^k}{k^k}$.

Suppose for games with k players, the first i players capture a minimum of $A_{[i]} \geq \frac{\sum_{j=0}^{i-1} \binom{i}{j} * k^j (-1)^{i-1-j}}{k^i}$ of what they optimally could. Now consider player

$i+1$. It captures a minimum of $a_{i+1} \geq \frac{1}{k}(1-A_{[i]})$ where

$$\begin{aligned}\frac{1}{k}(1-A_{[i]}) &\leq \frac{1}{k}\left(1-\frac{\sum_{j=0}^{i-1}\binom{i}{j} * k^j(-1)^{i-1-j}}{k^i}\right)\\ &= \frac{\sum_{j=0}^{i}\binom{i}{j} * k^j(-1)^{i-j}}{k^{i+1}}.\end{aligned}$$

We then have $A_{[i+1]} = A_{[i]} + a_{i+1}$, and note that the minimum value of $A_{[i+1]}$ can occur only if the minimum value of $A_{[i]}$ occurred (and thus we have the maximum guarantee on the value of a_{i+1}) as an increase of δ to $A_{[i]}$ results in a decrease of only $\frac{\delta}{k}$ to the guaranteed minimum of a_{i+1}. Thus

$$\begin{aligned}A_{[i+1]} &= A_{[i]} + a_{i+1}\\ &\geq \frac{\sum_{j=0}^{i-1}\binom{i}{j} * k^j(-1)^{i-1-j}}{k^i} + \frac{\sum_{j=0}^{i}\binom{i}{j} * k^j(-1)^{i-j}}{k^{i+1}}\\ &= \frac{\sum_{j=0}^{i}\binom{i+1}{j} * k^j(-1)^{i-j}}{k^{i+1}},\end{aligned}$$

which gives $A_{[k]} \geq \frac{\sum_{j=0}^{k-1}\binom{k}{j}*k^j(-1)^{k-1-j}}{k^k} = \frac{k^k-(k-1)^k}{k^k}$. As $k \to \infty$, we have

$$\begin{aligned}\lim_{k\to\infty}\frac{k^k-(k-1)^k}{k^k} &= \lim_{k\to\infty} 1-\left(\frac{k-1}{k}\right)^k\\ &= \lim_{k\to\infty} 1-\left(\left(\frac{k}{k-1}\right)^k\right)^{-1} = 1-\mathfrak{e}^{-1} = \frac{\mathfrak{e}}{\mathfrak{e}-1}.\end{aligned}$$

We can show that this is a tight lower (upper) bound on the efficiency (PoA) of the reserved path policy by constructing examples of networks with these efficiencies (PoAs). We do so by creating a graph with k parallel paths of length k^{k-1} and add edges in a way so that each player collects the minimum it is guaranteed, and visits all paths. Figure 4.3 displays this PoA of $\frac{4}{3}$ for $k=2$, and Figure 4.6 displays a PoA of $\frac{19}{27}$ for $k=3$, but due to exponential size of these graphs in k we have not provided images here for $k \geq 4$. □

We now consider the unreserved path policy. From Theorem 9, we know that the k-player game under the unreserved path policy represents a game over a valid utility system when played simultaneously. Therefore, we know that in the simultaneous setting the PoA of the k-player game is at most 2. However, we do not have a theorem concerning the PoA of k-player leader-follower games over valid utility systems. Because of this, we establish a

Table 4.3: Reserved Path Bounds

i	Min Captured	Min Captured, $k=i$
1	$\frac{1}{k}$	1
2	$\frac{2k-1}{k^2}$	$\frac{3}{4}$
3	$\frac{3k^2-3k+1}{k^3}$	$\frac{19}{27}$
4	$\frac{4k^3-6k^2+4k-1}{k^4}$	$\frac{175}{256}$
...	...	...
k	$\frac{\sum_{j=0}^{k-1}\binom{k}{j}*k^j(-1)^{k-1-j}}{k^k}$	$\frac{k^k-(k-1)^k}{k^k}$

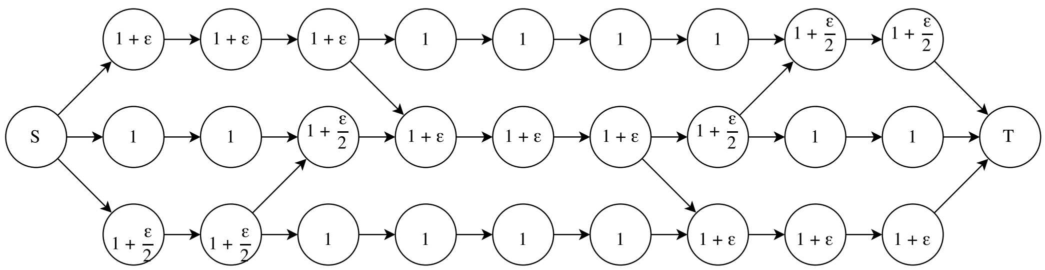

Figure 4.6: Network with a 3-Player PoA of $\frac{19}{27}$ for all policies

loose bound on the PoA of the unreserved path policy.

Lemma 7. *Under the unreserved path Policy, the Price of Anarchy for a k-player homogeneous game on a general graph is less than or equal to $\frac{k^2}{k-1}$.*

Proof. We will normalize $A^*_{[k]} = 1$. Consider the case where $k-1$ players have laid out their routes and collected a total of $A_{[k-1]}$. If $A_{[k-1]} \geq \frac{1}{k}$ then at least $A_{[k-1]}$ will be collected no matter what player k does. Therefore, assume $A_{[k-1]} < \frac{1}{k}$. By Lemma 6, there is a path q_k containing at least $\frac{1}{k}\left(1 - A_{[k-1]}\right) \geq \frac{k-1}{k^2}$ uncollected prizes, so we know that the k^{th} player will collect at least this many. □

Lemma 8. *Under all policies on an undirected graph with unlimited range, the general game has a PoA of 1.*

Proof. The reserved path policy is trivial: The first player will plot the shortest route which visits all nodes and collect all prizes. The unreserved path policy is similarly easy: The first player must plot a path which ultimately ends at t_1, as must the second player and so on. If by the time player k must select its route, there are still prizes which will remain uncollected, then player k will make sure to obtain all of them before terminating at t_k. The turn-based policy is slightly more difficult: Each player has determined its entire route prior to making its first move. Each infinite route is dominated by a finite route (unlike in the directed cyclic case seen in Figure 4.5) because after some time, none of the routes plan to collect anymore prizes. Since all players plan to end at t_i in a finite number of turns, some player i plans to finish last. i will not move to T until all remaining prizes are collected, because range is unlimited and the undirected graph has full bi-directional connectivity between all collectible prize nodes.

In the case where the graph has disconnected components, each component represents its own subgame where the above scenarios play out. □

4.5 Solution Methods

Having introduced the PCMOP and providing theoretical analysis of the PoA under our three policies in Sections 4.2 through 4.4, we next develop solution methodologies to solve a 2-player games with integer length edges. This section illustrates how to solve a game for each policy on directed acyclic graphs, beginning with an integer program formulation for the original TOP. Formulating the TOP is necessary for two reasons: First, to measure the PoA of a given problem to compare to our theoretical bounds, we must compute the optimal solution to the TOP. Second, all of our policy solution methods incorporate the TOP formulation as a helper function. Our reserved path solution method uses the TOP formulation repeatedly to iteratively construct each player's path, the solution method for the unreserved path approach uses it to iterate over multiple high value paths and expands it to compute best responses for the second player to the first player's path, and our solution method for the turn-based policy, which iterates over the game tree, will use the TOP formulation to avoid non-optimal leaves in the tree. All algorithms will be discussed in depth in the remainder of this section, with pseudocode for each provided in Appendix C.1.

TOP Formulation: We begin by formulating the TOP because each of our solution methods will either build on it or solve the formulation as a helper method. We model the problem as an Integer Program patterned on the integer-valued flow-conservation formulation given in [85], given here as IP1 in Appendix C.1. x_{ij}^m is binary to indicate whether agent m traversed edge (i, j), $z_i \in \{0, 1\}$ indicates whether the prize at node i is collected, l_{ij} is the length of an edge, and r is the common maximum range for all players. Finally, $d_j = 0$ except for $d_1 = 1$, $d_n = -1$ to establish the flow conservation constraints, and c_i is the value of the prize at node i.

Reserved Path Analysis: The reserved path policy is the simplest policy to derive a solution for. Because each player does not need to be concerned with the actions of any later player, each need only solve a single-player orienteering problem where the prizes that earlier players will collect are set to zero. Pseudocode is given in Algorithm 3 in Appendix C.1, where SolveIP1 is a helper function which solves the integer program formulation (IP1) of the TOP presented in the previous subsection.

Unreserved Path Analysis: Given the complexity of this problem, we only consider the two-player game. We consider the unreserved path as a two-stage game tree. While the players may no longer be able to observe each other after they begin moving through the network, each is required to declare its path prior to setting out. However, for a directed acyclic graph of n nodes there may be as many as 2^{n-2} paths going from node 1 to node n, meaning the tree may have as many as $2^{2(n-2)}$ leaf nodes. To reduce computation time we make the following observation: if the first player has found a path that gives v in prizes after the second player makes its best response, then any path containing less than v total prizes must be strictly worse for the first player than the path it has already found. Thus we can order the paths in the network according to the value of their prizes and stop searching paths once their value drops below the current best value v that the first player is able to obtain after player 2's best response. We refer to v and the corresponding path as the first player's best strategy so far. Therefore, we adopt a computation strategy based on computing the k'-best paths. Note that k' is distinct from k, the number of players in the game.

In order to compute the second player's best response, we use the integer program IP2 in Appendix C.1, where M is defined to be a large, positive constant. t_i^j is a variable that represents the time-step at which player j arrives at node i, and is 0 if i is not visited by j. v_i^j is a binary variable which is 1 if and only if player j visits node i. Because we are calculating a best response to player 1's path, all t_i^1 and v_i^1 are set to their appropriate value to represent the path, and are capitalized to T_i^1 and V_i^1 to indicate they are constants. We offset arrival times for visited nodes so that if the first player reaches node i at time j, then $T_i^1 = j - 0.5$, in order to represent that the first player obtains the prize in any tied arrivals. Again, it is important note that IP2 is made for Directed Acyclic Graphs, as constraint (C.13) will allow a node to be visited by the second player no more than once.

The problem of finding the top k' paths through a network is well studied, with efficient algorithms proposed as far back as the 1970's in [86] and [87]: these still form the basis for many algorithms used today. While these approaches are developed towards finding the k'-shortest or cheapest paths, it is possible that algorithms patterned after them may be developed in the case of directed acyclic graphs. In an enterprise-level solution this should be attempted, but we chose to solve the TOP iteratively with $k' = 1$ for convenience as we already developed the framework (IP1) to do so by setting $k = 1$. Paths are constructed iteratively, then compared the value of the path to the value v of the first player's current best strategy. If the value of the current path is less, we terminate and return the current best strategy. Otherwise, we compute the second player's best response to the path using IP2. Following the best response of the second player, the first player's best strategy is updated if a new best strategy is found. We then add a constraint to IP1 to disallow the current path, and continue to the next iteration. The pseudocode in Algorithm 4 in Appendix C.1 illustrates this method, where SolveIP2 computes the second player's best response to the first player's current path.

Turn-Based Analysis: For the turn-based policy, we again consider the game tree. The game tree has the same number of leaf nodes as the two-level game tree from the unreserved path policy, but resists an easy ordering of them. However, there is a great deal of commonality between several nodes in the game tree: For a DAG, if the next player to take a turn is at node i and the other player is at node j, then it does not matter where they were before, only which prizes have been collected on the nodes between i and j and the remaining range of each player. Because of this, we maintain a hashset with states of the game as keys, and the next move (i.e., next state) along with the value for each player of the current state. If a state is not already in the set then it has not been solved yet, and its solution is the successor state that leads to the maximum value. The recursion stops when it reaches a state where one player is at n. It then solves IP1 with $k = 1$ starting from the position of the unfinished player and returns. Algorithm 5 illustrates this approach, using Algorithm 6 as a helper function. The helper function fills in the game tree (StateSpace in Algorithm 5) for each feasible pair of positions for the two players, including which nodes between their positions have been visited. Finally, a second helper function ConstructPaths follows the pointers in the game tree to return the path of each player as a sequence of nodes. Pseudocode versions of both algorithms are given in Appendix C.1.

Of our solution methods, the turn-based solution is the only one which requires integer length edges. To represent the fact that players are moving and making decisions near simultaneously, we convert the integer edge-length graph into an unweighted graph, dividing a length l edge into l length-1 edges with $l-1$ nodes between them, containing no prizes.

4.6 Numerical Analysis

This section presents numerical experiments using the solution methods detailed in Section 4.5. Section 4.6.1 details our generation of test-cases for these methods, while Section 4.6.2 presents the results of games played on these test cases.

4.6.1 Test-Case Generation

Approximate $\mathcal{R}^2$ Networks

Because our motivation for considering the PCMOP is for its applications to UAV surveillance networks, we chose to generate networks which resemble real geography. We did so as follows: consider a $l \times l$ box in the plane ($\mathcal{R}^2$). To generate an n node DAG, place n points uniformly at random within the box and label the points so that $x_1 \leq x_2 \leq ... \leq x_n$. Compute the Euclidean distance between each pair of points. Let $D \in \mathcal{Z}^+$ be the maximum length permitted for any edge within the network. For $i < j$, if the distance from point i to point j is $dist(i,j) \leq D$, add an edge from node i to j of length $\lceil dist(i,j) \rceil$ to the DAG. We take the ceiling of the distance because, as noted in the previous section, it is easier to compute the turn-based game with integer length edges. Because we only consider adding an edge (i,j) if $i < j$, the resulting network will be a DAG. Having constructed the network, we then ran tests using two different prize distributions for the nodes: In the first, each node is assigned a prize drawn from the geometric distribution with $p = \frac{1}{2}$. In the second, each node is assigned a prize of value 1, 2, or 3, uniformly at random. Therefore both distributions generate the same mean prize value. Additionally, the prizes on $S = 1$ and $T = n$ are set to 0.

It should be noted that the resulting network may not have a path from 1 to n, or may not have one of sufficiently short length. In this case, a new network is generated.

Planar Networks

We were also interested in studying planar networks, as they are another common type of real network in $\mathcal{R}^2$. In particular, they better model a surveillance fleet which is restricted to roads. In order to generate these networks we again generate a set of n nodes uniformly in a box on the plane ($\mathcal{R}^2$). We then use the method of Delaunay triangulation to produce planar networks from these points. If an edge exists between nodes i and j such that $i < j$ then the edge is assigned to be (i, j) rather than (j, i), which guarantees the resulting network is a DAG. After constructing the network, we use the same uniform prize distribution we used previously and assign all nodes other than $S = 1$ and $T = n$ a prize of 1, 2, or 3 uniformly at random. We chose not to run trials for the geometric prize distribution because while we are interested in planar networks as a subset of networks in $\mathcal{R}^2$, they are less applicable to our use case as UAVs are not bound to existing roadways.

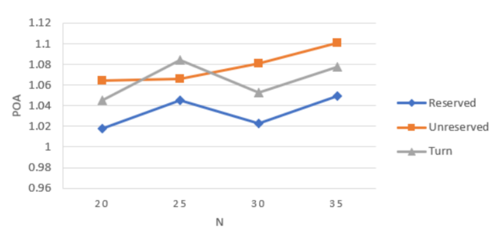

Figure 4.7: Average PoA for Approximate $\mathcal{R}^2$ Networks with Geometric Prize Distribution

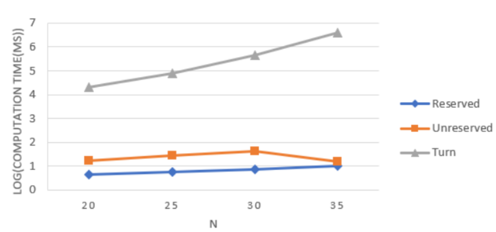

Figure 4.8: Average Computation time (log-scaled) for Approximate $\mathcal{R}^2$ Networks With Geometric Prize Distribution

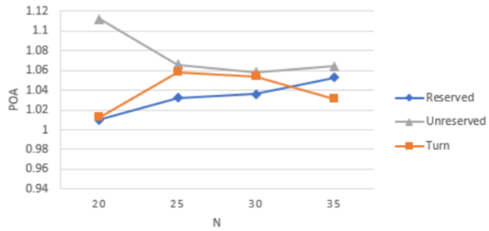

Figure 4.9: Average PoA for Approximate $\mathcal{R}^2$ Networks with Uniform Prize Distribution

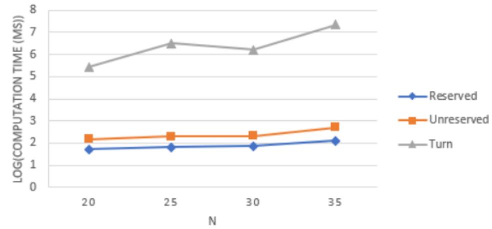

Figure 4.10: Average Computation time (log-scaled) for Approximate $\mathcal{R}^2$ Networks With Uniform Prize Distribution

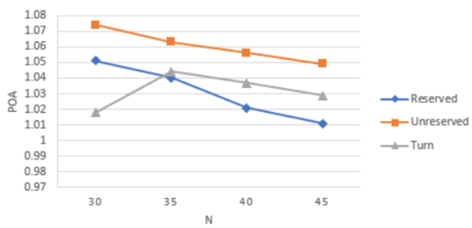

Figure 4.11: Average PoA for Planar Networks with Uniform Prize Distribution

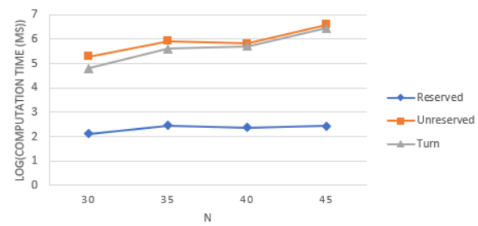

Figure 4.12: Average Computation time (log-scaled) for Planar Networks with Uniform Prize Distribution

4.6.2 Empirical Results

Table 4.4 at the end of this section presents the full details of our numerical experiments, both in our approximate $\mathcal{R}^2$ and planar networks. Max Edge refers to the maximum length edges inserted into the graph as described in Section 4.6.1. Box refers to the edge length l of the box used to generate the graph. All entries related to Computation Time are in seconds. Tests for uniformly distributed prizes in both planar and approximate $\mathcal{R}^2$ networks were run on machines using the Windows 10 Enterprise OS and 16GB RAM with Intel Xeon(R) v6 CPUs at 3.30GHz. Tests for the geometrically distributed prizes for the approximate $\mathcal{R}^2$ networks were run with Windows 10 Pro OS and 32GB RAM with Intel(R) i7 CPUs at 2.6GHz.

To test how PoA changes with the average size of the network, we generated instances of approximate $\mathcal{R}^2$ networks with varying numbers of nodes n. These trials are detailed in Table 4.4. Additionally, Figures 4.7 and 4.9 show the average PoA for approximate $\mathcal{R}^2$ networks as function of n when these graphs are generated in an $l = 10 \times 10$ box, with max edge length 3, range 15, and prizes drawn geometrically with $p = \frac{1}{2}$ and uniformly at random from 1, 2, 3, respectively. Figures 4.8 and 4.10 show the corresponding average computation times for solving these networks, given as the log of the average milliseconds (ms) required.

When we consider the figures, we see that the computation times for the two prize distributions display similar behavior, with an approximately linear increase in log-scaled mean computation time as a function of the number of nodes n. The unreserved case for $n = 35$ and geometrically distributed prizes is the main contradiction to this statement, however this is likely due to the fact that only 5 trials were run for this instance: Considering the high sample variance for the unreserved policy when $n = 25, 30$ when compared to their sample means, it is evident that when test cases are generated according to this distribution there is a tendency to produce very difficult outlying instances, which may not occur with a smaller sample size of test cases.

The more interesting figures to consider are Figures 4.7 and 4.9. While test cases generated from the uniform distribution in most cases demonstrate a lower average PoA, the results are empirically very close. However, while we see a possible upward trend with respect to n in the average PoA for all policies when prizes are geometrically distributed, the same is not true of the networks when prizes are uniformly distributed: While the reserved policy seems to display an upward trend, it is difficult to say what if any regular behavior the average PoAs of the unreserved and turn-based policies display.

While we are interested in planar networks as they are an important subset of graphs in $\mathcal{R}^2$, we have already noted that they are less relevant to our UAV use-case as UAVs are not bound to existing infrastructure. Because of this, and the fact that computing the equilibrium in the turn-based game is the most computationally expensive portion of the approximate $\mathcal{R}^2$ networks, we used our planar test cases to consider what happens to the turn-based game when some of the computational complexities disappear. In particular, all tests were run with unit edge lengths and unlimited range, with prizes again drawn uniformly at random from 1, 2, and 3 for each node. The resulting average PoA's as a function of n can be seen in Figure 4.11, with Figure 4.12 containing the associated average computation times. Unsurprisingly, this reduces the computation time immensely for the turn-based game, bringing computation time down by a factor of approximately e compared to a uniform distribution of prizes over a similarly sized approximate $\mathcal{R}^2$ network which can be seen by comparing Figures 4.10 and 4.12. What is initially surprising though is the degree to which it raises the computational effort for calculating the unreserved equilibrium. However, upon further consideration the reason becomes clear: As what is essentially a bi-level optimization problem in which the first player needs to calculate its optimal move given the best response of the second player, removing limits on the range of each player, even in a DAG, allows for the potential of exponentially more strategies being available without a method to remove consideration for some of these strategies. Still, the fact that the average computational effort exceeds that of the turn-based game is noteworthy. The variance in computation times sheds some light on this, as Table 4.4 shows in all cases the variance in computation time for the unreserved case is an order of magnitude or more higher than the variance for the turn-based, suggesting that some outlying test networks are particularly difficult to solve.

Table 4.4: Numerical Results related to PoA and Computation Times

$\mathcal{R}^2$ Approximate Networks Geometric Prize Distribution					PoA						Computation Time (s)					
					Reserved		Unreserved		Turn		Reserved		Unreserved		Turn	
N	Box	Max Edge	Range	Trials	Mean	Var	Mean	Var	Mean	Var	Mean	Var	Mean	Var	Mean	Var
20	10	3	15	5	1.018	.002	1.064	.012	1.045	0.004	0.042	1E-4	.168	.056	201.56	1.30E5
25	10	3	15	15	1.045	.004	1.066	.008	1.084	.006	.056	3E-4	.280	.474	763.14	1.57E6
30	10	3	15	15	1.023	.001	1.081	.007	1.053	.002	.074	4E-4	.413	.287	4.53E3	1.18E8
35	10	3	15	5	1.049	.001	1.101	.007	1.078	.007	.097	4E-4	.159	.006	4.06E4	4.85E9
20	7	2	12	5	1.018	6E-4	1.12	.022	1.038	.002	.031	1E-4	.186	.027	57.839	5.85E3
30	5	2	∞	2	1.009	1E-4	1.074	.011	1.06	.003	.109	5E-10	329.5	1.39E5	1.05E6	1.43E10

$\mathcal{R}^2$ Approximate Networks Uniform Prize Distribution					PoA						Computation Time (s)					
					Reserved		Unreserved		Turn		Reserved		Unreserved		Turn	
N	Box	Max Edge	Range	Trials	Mean	Var	Mean	Var	Mean	Var	Mean	Var	Mean	Var	Mean	Var
20	10	3	15	5	1.010	5E-4	1.112	.040	1.013	.001	.054	2E-4	.154	.006	285.30	1.71E5
25	10	3	15	15	1.032	.002	1.066	.005	1.058	.006	.066	4E-4	.201	.071	3.27E3	1.02E8
30	10	3	15	15	1.036	.003	1.058	.004	1.054	.007	.072	3E-4	.215	.051	1.63E3	1.46E7
35	10	3	15	5	1.053	.004	1.064	.007	1.031	.005	.128	.001	.503	.076	2.23E4	2.50E8
20	7	2	12	5	1.020	.001	1.024	.001	1.012	.001	.044	1E-4	.425	.258	69.074	1.25E4
30	5	2	∞	2	1.010	2E-4	1.039	.001	1	0	.115	.001	756.35	1.00E6	2.91E4	8.90E8

Delaunay Triangular Networks					PoA						Computation Time (s)					
					Reserved		Unreserved		Turn		Reserved		Unreserved		Turn	
N	Box	Max Edge	Range	Trials	Mean	Var	Mean	Var	Mean	Var	Mean	Var	Mean	Var	Mean	Var
30	N/A	1	∞	10	1.051	.001	1.074	.004	1.018	.001	.130	.001	191.84	9.26E4	62.84	596.8
35	N/A	1	∞	10	1.040	.003	1.063	.002	1.044	.002	.283	.005	842.78	3.26E6	406.68	4.22E4
40	N/A	1	∞	10	1.021	.001	1.056	.001	1.037	.001	.234	.007	668.75	1.84E6	509.46	1.09E5
45	N/A	1	∞	5	1.011	1E-4	1.049	.003	1.029	.002	.268	.004	4.00E4	1.79E9	2.82E3	8.54E6

4.7 Summary and Discussion

In this paper, we introduced the Prize-Collecting Multi-Agent Orienteering Problem and proposed three policies to govern the selfishness of the agents. Given that the PCMOP lies at the intersection of congestion games, shortest path computation, longest path computation, top k paths computation, and the TOP, it is natural that it inherits complexities from each of them, in particular an extreme sensitivity to changes in parameter values. Despite these complexities, we derived theoretical bounds on the maximum inefficiency possible under each of these policies in the form of PoAs. As part of that analysis, we extended [2]'s result related to PoA of games over valid utility systems to a 2-player leader-follower setting. In addition to theoretical bounds, we developed solution methods to solve a PCMOP under three policies. In terms of empirical efficiency, there are relatively small differences in the average PoA of the three policies, as seen in Figures 4.7, 4.9, and 4.11. While the reserved path policy produces the best average PoA in most cases, we see there are some where the average performances of the turn-based policy surpasses it. Also, while the performance of the unreserved policy is the worst on average for all but one of our test cases, there are individual instances where it delivers the best value, albeit not as many as the reserved and turn-based policies.

We have seen that the reserved path policy has the best theoretical guarantees on performance in terms of prize-collecting. However, it may produce poorly distributed prize divisions: consider that with sufficient range on an undirected graph, the first agent will collect all prizes, which defeats the purpose of using multiple agents. Also, while the reserved path policy has the best *guarantees*, it does not always produce the best *results*: A 2-Player game on the directed graph in Figure 4.13 using either the unreserved or turn-based policies results in all prizes being collected, but the reserved path policy results in a PoA of $\frac{4+2\varepsilon}{3+2\varepsilon}$, its theoretical worst result.

The unreserved path and turn-based policies appear to produce the best results in terms of distributions: earlier players have a hierarchical advantage, being able to get any specific prize they want before or at the same time as lower-ranked players and taking it from them, but they must weigh the cost of the prizes they can no longer guarantee themselves against the ones they want. The turn-based policy seems particularly realistic: if an earlier player

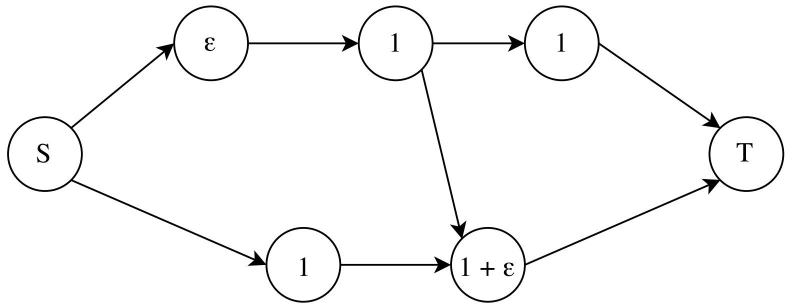

Figure 4.13: Network with a 2-Player PoA of $\frac{4}{3}$ for reserved policy and 1 for other policies

makes it clear it is going after a specific large prize, it can make others back off sooner and allow the player to stop and collect additional smaller prizes along the way, rather than racing by them (as in the unreserved policy) even though no other player is targeting the same prize. This is seen in Figure 4.1, where the prize at node B is collected in the turn-based policy but not the unreserved path policy, resulting in a PoA of $\frac{4}{3}$ for the unreserved case.

The main drawback to the turn-based policy is the computational complexity: The game tree is exponentially large and while there is frequent similarity among branches to reduce computations, it is still a problem. A one player PCMOP/TOP is solved at each leaf node which is not recognized as already solved, which, although likely a smaller problem (since much of the graph is already traversed), is NP-Hard.

The advantage of the reserved path policy is that it is relatively easy to compute, as it allows earlier players to completely ignore the actions of later players, and every player after the first can remove prizes from the network and pretend that it is the first player (for computation). Then each agent only has to solve the single player PCMOP/TOP which, while still NP-Hard, is more tractable than the other versions. While it can cause unbalanced prize distributions in general, it is unlikely to do so when agent ranges divides the workload approximately evenly between agents.

The unreserved path policy seeks to reduce the computational effort of the turn-based policy by considering the best paths first: while computation tends to be longer when there are a number of paths with the same value in prizes, such as when the range is relatively large, in most cases the runtime is shorter than in the turn-based policy. It also appears to produce a more even distribution of prizes than the reserved path policy.

CHAPTER 5

DIRECTIONS FOR FUTURE RESEARCH & PROGRESS SINCE PRELIMINARY EXAM

In this chapter we will discuss progress which has been made in this dissertation since it was first presented in my doctoral exam, as well as propose future directions to take the research contained here.

The most visible progress made has been the completion of Chapter 3. While the bones of the chapter and most of its mathematical formulae and theorems were already present, they were in a far rougher form which was unsuitable for publication. Further, numerical studies at that point in time were minimal. The majority of the time which has passed since my preliminary exam in May 2020 has been spent experimenting with this model on real social networks. While unexpectedly time-consuming, this led to a wealth of evidence in support of our model from the fields of evolutionary biology and psychology, as well as the surprising result of the difference between lexicographic and random tie-breaking (this was discovered entirely by accident thanks to a coding mistake.) It also led to the surprising discovery that increasing the opportunities available to agents (number of invitations k) actually led to a decrease in trustworthy behavior rather than an increase as we had initially expected.

While resulting in less change to this dissertation, two other accomplishments which are just as important if not more are the acceptance for publication of work based on Chapter 2 and Chapter 4. Work [3] based on Chapter 2 was accepted for publication in the *European Journal of Operational Research* in July 2020 and will be published in February 2021, however it is already available online. Work based on Chapter 4 was accepted more recently to *IEEE Transactions on Automation Science and Engineering* in late October 2020 and thus is not available online at the time of writing.

Finally, I have completed additional work on the topic of minimizing the spread of an influence cascade through counter-cascades which I proposed at the time of my preliminary examination. It is currently being prepared for a conference submission. This work is independent of the work work presented in this dissertation, both in the thematic sense as it is not connected to the topic of trust and minimizing selfishness, and in the literal sense as I have completed it independently from my advisors who have contributed to all of the work presented here. As such it is not included in this dissertation.

Next, we will discuss future research directions for the work presented in this dissertation. There are two major directions for the work in Chapter 3. The first is a study of network structure and how it influences behavior in the system, as we saw that it does indeed have a strong impact on how selfish agents decide to be. The second more theoretical route involves as study of the metagame in Chapter 3.6, as at this point in time we have only been able to prove the existence of a Nash equilibrium and nothing else. There is also a third, heavily applied, direction. This would be the development of a bot to act as an agent in this system through a method such as reinforcement learning. I had initially planned to include work on this topic in this dissertation, but due to the extensive and time-consuming numerical studies in this work I was unable to.

There is also continuing research to be done on the work in Chapter 4. The motivation of that chapter was to find a policy that forced agents to divide up tasks "fairly" and with minimal inefficiency. One area which we are currently just beginning to pursue is the study of fair-division in chores. This area has received attention in recent years, and building on results discovered in our own department I am interested in methods which fairly allocate chores which have transition times between them. This general framework easily lends itself to the setting considered in Chapter 4 and I would be interested in comparing them. This work is still barely beginning.

CHAPTER 6

CONCLUSION

This chapter marks the end of this dissertation. Thus far, we have presented three major topics of research. The first, in Chapter 2, is the introduction of a model of non-selfish behavior called the limited-trust equilibrium. This concept captures the idea that agents are willing to help each other, provided it is not too expensive for them personally. They do so in order to inspire reciprocal behavior from each other later on, sacrificing a small gain in the present in the hope of avoiding a larger cost in the future. Following its theoretical analysis, we show numerically that when two agents play in a limited-trust manner both stand to gain in expectation.

The second topic of research builds upon the foundations and motivations of the first. When considering limited-trust players in a social network trying to form partnerships, we see empirically that not playing in a myopically self-interested manner in each game is actually the most selfish action that many agents can take. This is because it allows them to form more partnerships, and although these players derive less utility per partnership, they derive more by obtaining a greater volume. It also benefits their partners, as that is the reason these partnerships were formed. Interestingly, we also see that in this setting there is a metagame of selecting the right trust level in response to other players. This has the potential to be a rich area of both theoretical and applied research in the future, but for now we are able to show that this metagame is guaranteed to possess a mixed Nash equilibrium in certain circumstances.

The third major topic of research is a bit of a departure from the first two, proposing and considering a game theoretic version of the orienteering problem. This formulation is motivated largely by UAV fleet coordination, and considers the inefficiency of policies under which agents are allowed to route themselves, rather than consulting a central fleet manager. This research also extends a broad result from [2] on the price of anarchy of valid utility games to the leader-follower setting.

After presenting the core of the author's research during the graduate program, in Chapter 5 a discussion of changes from the version of this dissertation presented in the Preliminary Exam is given, as well as additional research proposals for future work. These proposals include new directions to take from the results of Chapter 3, as well as a new method for approaching the research in Chapter 4 based on fair allocation of chores.

APPENDIX A

LIMITED-TRUST EQUILIBRIA

A.1 Proofs for Chapter 2

A.1.1 Proof of Lemma 1

Proof. Consider that

$$\sigma_i^* = \arg\max_{\sigma_i \in \Sigma_i} u(\sigma_i, \sigma_{-i}^*)$$

subject to

$$\delta_i \geq u_i(\sigma_i^G, \sigma_{-i}^*) - u_i(\sigma_i, \sigma_{-i}^*)$$

prior to the constant c_j being added. Each player $i \neq j$ now has to solve

$$\sigma_i^{*\prime} = \arg\max_{\sigma_i \in \Sigma_i} (u(\sigma_i, \sigma_{-i}^*) + c_j)$$

subject to

$$\delta_i \geq u_i(\sigma_i^G, \sigma_{-i}^*) - u_i(\sigma_i, \sigma_{-i}^*)$$

which has the same solution, and player j has to solve

$$\sigma_j^{*\prime} = \arg\max_{\sigma_j \in \Sigma_j} (u(\sigma_j, \sigma_{-j}^*) + c_j)$$

subject to

$$\delta_i \geq (u_j(\sigma_j^G, \sigma_{-j}^*) + c_j) - (u_j(\sigma_j, \sigma_{-j}^*) + c_j)$$

which also has the same solution. Therefore, σ^* is still an LTE(δ). □

A.1.2 Proof of Theorem 6

Proof. First, suppose that we have a feasible solution (x, y). Given that it is a feasible solution, the last four constraints will not be violated as x and y are valid strategy profiles. The first two constraints ensure that neither player is giving up more than δ_i that it could be making by playing the greedy best response to the other player's strategy. The next two ensure that the social utility from the players actions is at least the amount which would be provided if each player played its limited trust best response to the other. Therefore, since each player is providing as much social utility as if it had been playing its limited trust best response (more is impossible without violating constraint 1 or 2) and is not giving up more than δ_i of what it could be making, (x, y) is an LTE(δ).

Now, consider an LTE(δ) given by (x, y), which we will show to be a feasible solution. Because (x, y) is an LTE, x and y are both valid strategy profiles and thus do not violate constraints (5-10), particularly as constraints (7) and (8) are disabled for $S_A = [m]$, $S_B = [n]$. (x, y) is an LTE, so neither player is giving up more than δ_i and therefore constraints (1) and (2) are fulfilled. Finally, because it is an LTE, x is a limited-trust best response to y and y is a limited-trust best response to x, which means that constraints (3) and (4) are fulfilled with equality. $\square$

A.2 Additional Results: Leader-Follower Games

A.2.1 Demonstration of Policies

Given the interpretation of the policies in Section 2.4, we now provide a practical demonstration of each on the 2×2 game given in Table A.1. The Stackelberg equilibrium in this game occurs when the first player plays 2 and the second player plays 1. We first consider the incomplete knowledge policy: for $\delta_1 < 1$, the first player will play 2, as it otherwise stands to lose $5 - 4 = 1$ if it plays 1. If the first player plays 1, then for $\delta_2 < 1$, the second player plays 2, but for $\delta_2 \geq 1$, the second player plays 1 as well. If the first player plays 2, then the second player plays 1 for $\delta_2 < 1.5$ and 2 for $\delta_2 \geq 1.5$. Thus for the socially optimal policy (1,1) to be played, we must have $\delta_1, \delta_2 \geq 1$.

Table A.1: A 2-Player 2×2 Game

		p_2 1	2
p_1	1	6, 4	4, 5
	2	5, 3	8, 1.5

We next consider the complete knowledge policy. The second player's behavior is the same: if the first player plays 1 the second plays 1 for $\delta_2 \geq 1$ and if the first player plays 2 the second plays 2 for $\delta_2 \geq 1.5$. Suppose $\delta_2 < 1$, then the first player is selecting between (2,1) and (1,2): if $\delta_1 \geq 1$ then it plays 1, and otherwise it plays 2. Suppose $1 \leq \delta_2 < 1.5$, then the first player is choosing between (1,1) and (2,1) and so plays 1 regardless of δ_1. Finally, suppose $\delta_2 \geq 1.5$. Here, the first player is selecting between (1,1) and (2,2). If $\delta_1 < 2$, the first player plays 2 to receive 8 after the second player also plays 2. If $\delta_1 \geq 2$ it plays 1 instead.

The complete knowledge policy has the property that if we have $(\delta_1', \delta_2') \geq (\delta_1, \delta_2)$, that does not mean that the net utility for the (δ_1', δ_2') game is greater than or equal to that of the (δ_1, δ_2) game: the $\delta_1 = \delta_2 = 1$ game results in a greater net utility than the $\delta_1 = \delta_2 = 1.5$ game over the utility above. This is interpreted as the first player taking advantage of the second player's trustworthiness. If $(\delta_1', \delta_2') \geq (\delta_1, \delta_2)$ then we can say that at least one of the following is true: player 1 receives higher utility under (δ_1', δ_2'), or the net utility is greater under (δ_1', δ_2'). This is because had player 1 selected the same strategy as it would with (δ_1, δ_2), it would have achieved at least as much utility for $\delta_2' \geq \delta_2$. The fact that it did not select the same strategy indicates that it selected one which increases either its own utility or the net utility.

Finally, we consider the cooperative complete knowledge policy. This policy is especially interesting as for any $\delta_1, \delta_2 \geq 0$, the social optimum of (1,1) will be played: as stated earlier, the Stackelberg equilibrium in this game occurs when the first player plays 2 and the second player plays 1. The social optimum occurs at (1,1), with a net payoff of 10, and at the social optimum both players are receiving strictly more than they would in the Stackelberg equilibrium. This is the only policy which has the possiblity of reward without risk: for any game in which $\delta_1 = \delta_2 = 0$, each player is guaranteed a minimum of what they would achieve in the Stackelberg equilibrium and the possibility of more. For the other two policies, $\delta_1 = \delta_2 = 0$ guarantees that they will play the Stackelberg equilibrium, meaning no risk, no reward.

Because the behavior of games under the cooperative complete knowledge policy is different from traditional leader-follower games even when $\delta_1 = \delta_2 = 0$, we will use the term *zero-trust* game to refer to the traditional leader-follower game.

A.2.2 Playing Social Optima

In this subsection we will examine what the structure of a 2-player bimatrix game must be in order for the Socially Optimal (greatest net payoff) strategy combination to be played. We will confine our discussion to 2×2 games, but the result generalizes to $m \times n$ games.

Let A, B be the payoff matrices for the first and second players, respectively. Without loss of generality, assume that the social optimum occurs when (1,1) is played.

First, we consider what must happen for the Stackelberg equilibrium to be the social optimum. An immediate requirement is $b_{12} \leq b_{11}$, as otherwise even if the first player plays 1, the second player will play 2. Without loss of generality, assume $b_{21} \geq b_{22}$ so that if the first player plays 2, the second player will play 1. In order for the first player to play 1 rather than 2 we must have $a_{11} \geq a_{21}$. More formally, (1,1) is the social optimum and gets played if and only if E_1, E_2, E_3, E_4 are satisfied where

$$
\begin{aligned}
E_1 &= (a_{11} + b_{11} \geq \{a_{12} + b_{12}, a_{21} + b_{21}, a_{22} + b_{22}\}) \\
E_2 &= (b_{2i} \geq \{b_{21}, b_{22}\}) \\
E_3 &= (a_{2i} \leq a_{11}) \\
E_4 &= (b_{11} \geq b_{12}).
\end{aligned}
$$

and i is player 2's best response to the first player playing 2.

Geometrically, we can determine the outcome of a game by plotting its entries in $\mathcal{R}^2$ in terms of the utility for each player, as in Figure A.1 where the horizontal axis a is the utility for the first player and the vertical axis b is the utility for the second player.

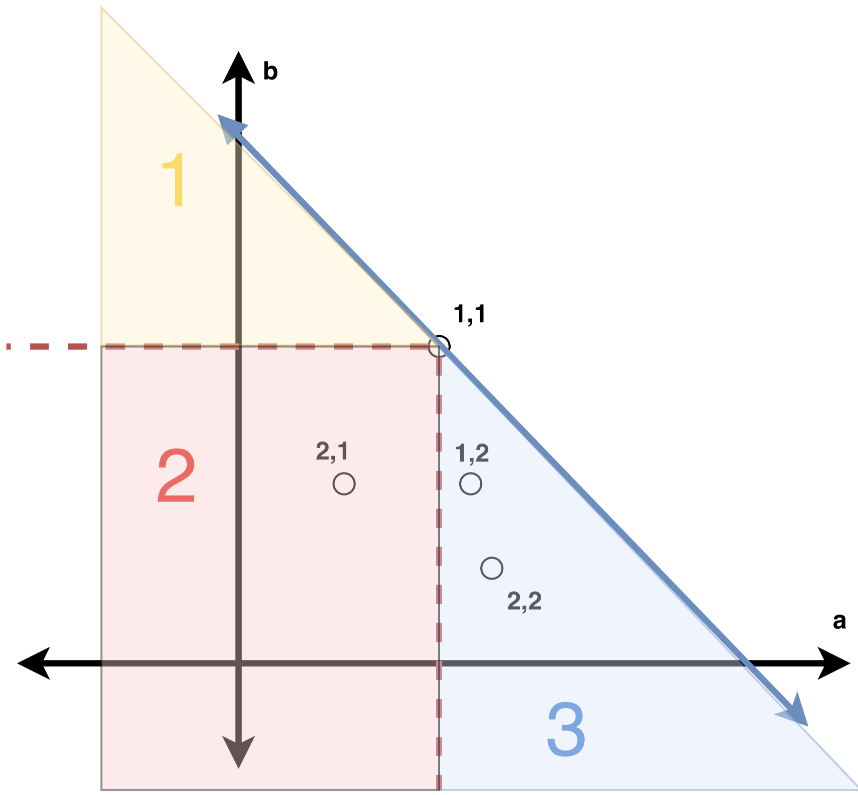

Figure A.1: Geometric Leader-Follower Representation

From the figure, we can see that (1,1) is the social optimum. It is also the Stackelberg equilibrium: $b_{12} < b_{11}$, so the second player will play 1 if the first plays 1. $b_{21} > b_{22}$, so the second player will play 1 if the first plays 2, and $a_{21} < a_{11}$, so the first player will play 1. In terms of the geometry, we can say that (1,1) is the Stackelberg equilibrium and the social optimum if and only if (1,2) is not in area 1 and whichever is larger out of b_{22} or b_{21}, that point is not in area 3. Then the first player will prefer 1 to 2 and the second will prefer (1,1) to (1,2). This is true for all complete knowledge 2×2 leader-follower games, and is easily generalized to $m \times n$ games.

Now consider the incomplete knowledge policy. If the social optimum occurs at (1,1), what must occur for it to be played? If $b_{11} \geq b_{12}$ and $b_{2i} \geq b_{21}, b_{22}$ the first player will play 1 if $a_{2i} \leq a_{11} + \delta_1$, and the second player will also play 1. If $b_{11} < b_{12}$ and $b_{2i} \geq b_{21}, b_{22}$, then the first player will play 1 if $a_{2i} < a_{12} - \delta_1$ or if $a_{2i} \leq a_{12} + \delta_1$ and $a_{2i} + b_{2i} < a_{12} + b_{12}$ or $a_{2i} \geq a_{12} + \delta_1$. If the first player plays 1, the second player will play 1 as well if $b_{12} \leq b_{11} + \delta_2$. We can therefore say that (1,1) is the social optimum and gets played if and only if E_1, E_2, F_3, E_4, E_1, E_2, F_5, F_6, F_7, or $E_1, E_2, F_5, \neg F_6, F_8$ are satisfied where

$$F_3 = (a_{2i} \leq a_{11} + \delta_1)$$
$$F_5 = (b_{11} < b_{12} \leq b_{11} + \delta_2)$$
$$F_6 = (a_{2i} + b_{2i} < a_{12} + b_{12})$$
$$F_7 = (a_{2i} \leq a_{12} + \delta_1)$$
$$F_8 = (a_{2i} \leq a_{12} - \delta_1)$$

As with the Stackelberg game, we geometrically model these constraints by plotting the payoffs in $\mathcal{R}^2$, with both possible constraint sets seen in Figures A.2 and A.3. On both figures, given the points $(1,1)$ and $(1,2)$, the social optimum is played if whichever is greater of b_{21}, b_{22} is in the shaded blue area.

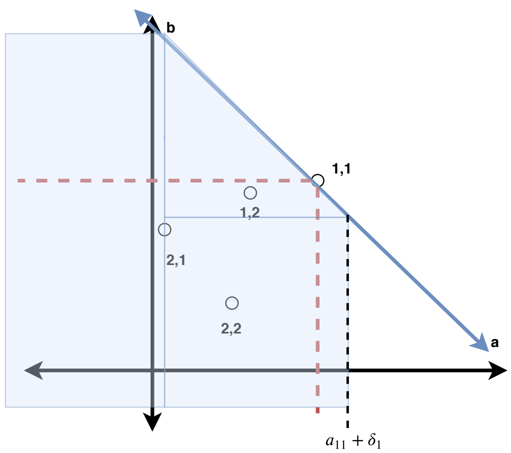

Figure A.2: E_1, E_2, F_3, E_4

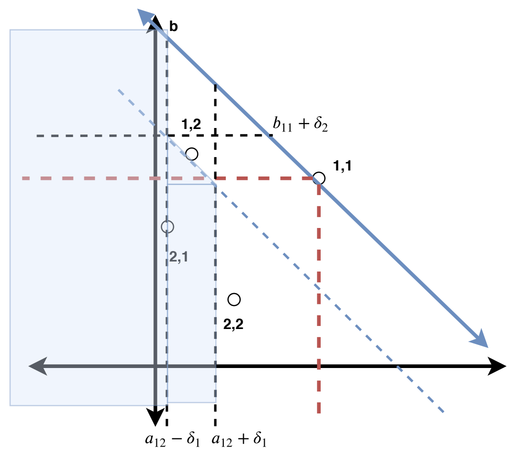

Figure A.3: E_1, E_2, F_5, F_6, F_7

We next consider the complete knowledge policy. Again we assume, without loss of generality, that (1,1) is the social optimum. Let $b_{2i} \geq b_{21}, b_{22} \geq b_{2j}$. We want to consider when the second player would play j given that the first player plays 2. This only occurs if $a_{2i} + b_{2i} < a_{2j} + b_{2j}$ and $b_{2i} < b_{2j} + \delta_2$. Now we can determine that the first player will only choose 2 (given the second player would play the social optimum if the first chose 1) if $a_{2i} > a_{11} + \delta_1$ or $a_{2j} > a_{11} + \delta_1$ and $a_{2i} + b_{2i} < a_{2j} + b_{2j}$ and $b_{2i} \leq b_{2j} + \delta_2$. Therefore, the social optimum gets played if and only if E_1, E_2, F_3, F_5 AND (G_9 OR G_{10} OR G_{11})

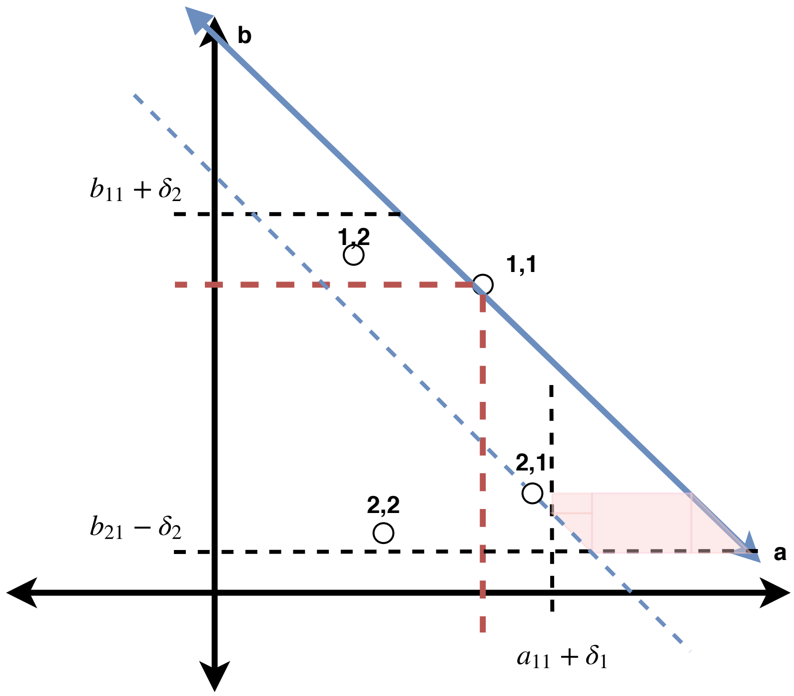

Figure A.4: Geometric Complete Knowledge Leader-Follower Representation

is satisfied where

$$G_9 = (a_{2j} \leq a_{11} + \delta_1)$$
$$G_{10} = (a_{2i} + b_{2i} \geq a_{2j} + b_{2j})$$
$$G_{11} = (b_{2i} \geq b_{2j} + \delta_2).$$

Suppose $b_{21} \geq b_{22}$. If the geometric representation of the incomplete knowledge policy primarily depended on (2,1) and was dictated by (1,2), here it is dependent on (2,2) and dictated by (2,1). Figure A.4 displays this: given the (1,1), (1,2), and (2,1) the the red region represents where the point (2,2) cannot be in order for the social optimum to be played. Additionally, we must have $b_{12} \leq b_{11} + \delta_2$ and $a_{21} \leq a_{11} + \delta_1$.

Finally, we consider the cooperative complete knowledge policy. It is easy to write the requirements for the social optimum to be played in terms of the Stackelberg equilibrium: if (1,1) is the social optimum and (i, j) is the Stackelberg equilibrium, (1,1) is played if and only if $a_{ij} \leq a_{11} + \delta_1$ and $b_{ij} \leq b_{11} + \delta_2$. Figure A.5 demonstrates this: if (1,1) is the social optimum, it

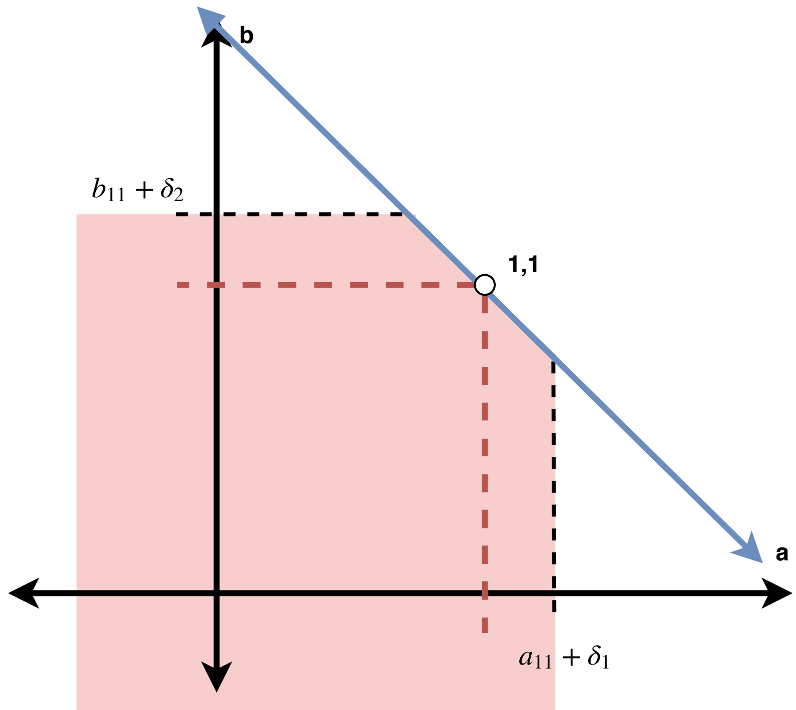

Figure A.5: Geometric Cooperative Complete Knowledge Leader-Follower Representation

is played if and only if the Stackelberg equilibrium occurs in the red shaded area.

A.2.3 Leader-Follower Random Repeated Games

While the LTE is capable of analyzing one-off games, it is also a suitable tool for repeated games with a particular emphasis placed on day-to-day societal interactions. Such interactions between players are not identical, but will likely display a pattern over time so that they could be said to come from some "typical" distribution. Because of this, we now consider games where the payoff matrices A, B for each player are generated according to some distribution. In particular, we consider matrices where each entry is generated iid for each player, though A and B may not come from the same distribution. Let f_a and f_b be the probability distribution functions of A and B, respectively.

Theorem 15. *In a 2×2 leader-follower game with zero trust where all of player 1's payoffs are a_{ij} iid and are independent from player 2's payoffs b_{ij}*

which are iid, the probability that Stackelberg equilibrium of the game is the social optimum is

$$P(S.O.) = 8\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} M(a_{11}, b_{11}) f_a(a_{11}) da_{11} f_b(b_{11}) db_{11}$$

where

$$\begin{aligned} M(a_{11}, b_{11}) = &\left(\int_{-\infty}^{a_{11}}\left(\int_{-\infty}^{a_{11}+b_{11}-a_{21}} G(a_{11}, b_{11}, b_{21}) f_b(b_{21}) db_{21}\right) f_a(a_{21}) da_{21}\right) \\ &\cdot\left(\int_{-\infty}^{b_{11}}\left(\int_{-\infty}^{a_{11}+b_{11}-b_{12}} f_a(a_{12}) da_{12}\right) f_b(b_{12}) db_{12}\right) \\ G(a_{11}, b_{11}, b_{21}) = &\left(\int_{-\infty}^{b_{21}}\left(\int_{-\infty}^{a_{11}+b_{11}-b_{22}} f_a(a_{22}) da_{22}\right) f_b(b_{22}) db_{22}\right) \end{aligned}$$

Proof. Given that every entry in the payoff matrix is equally likely to be the social optimum, and equally likely to be played if it is the social optimum, we will approach this problem by finding the probability that $1, 1$ is the social optimum, and is played. This occurs if E_1, E_2, E_3, and E_4 are satisfied. Because E_2 and E_3 are equally likely to be fulfilled by $i = 1$ or $i = 2$ and each is equally likely to occur, we can pick $i = 1$ and multiply by two. While it is easy to evaluate if any given game results in the social optimum being played, it is difficult to organize the integrals necessary to construct the probability of it occurring. To help visualize the integrals, we consider the graphical representation of a 2×2 game given in Figure A.1. For a fixed a_{11}, b_{11}, we have

$$\begin{gathered} P(a_{11} + b_{11} \geq a_{21} + b_{21}, a_{22} + b_{22}, E_2, E_3 | a_{11}, b_{11}) = Q(a_{11}, b_{11}) = \\ \int_{-\infty}^{a_{11}}\left(\int_{-\infty}^{a_{11}+b_{11}-a_{21}} G(a_{11}, b_{11}, b_{21}) f_b(b_{21}) db_{21}\right) f_a(a_{21}) da_{21} \end{gathered}$$

where

$$G(a_{11}, b_{11}, b_{21}) = \int_{-\infty}^{b_{21}}\left(\int_{-\infty}^{a_{11}+b_{11}-b_{22}} f_a(a_{22}) da_{22}\right) f_b(b_{22}) db_{22}$$

For 1,1 to be the social optimum and be played by player 2 if player 1 plays 1, it is necessary and sufficient that $b_{12} \leq b_{11}$ and $a_{11} + b_{11} \geq a_{12} + b_{12}$.

Since

$$P(a_{11} + b_{11} \geq a_{12} + b_{12}, b_{11} \geq b_{12} | a_{11}, b_{11}) = H(a_{11}, b_{11})$$
$$H(a_{11}, b_{11}) = \int_{-\infty}^{b_{11}} \left(\int_{-\infty}^{a_{11}+b_{11}-b_{12}} f_a(a_{12}) da_{12} \right) f_b(b_{12}) db_{12}$$

is independent of $P(a_{11} + b_{11} \geq a_{21} + b_{21}, a_{22} + b_{22}, E_2 | a_{11}, b_{11})$, we then have

$$\begin{aligned} P(E_1, E_2, E_4 | a_{11}, b_{11}) &= P(a_{11} + b_{11} \geq a_{12} + b_{12}, b_{11} \geq b_{12} | a_{11}, b_{11}) \\ &\quad \cdot P(a_{11} + b_{11} \geq a_{21} + b_{21}, a_{22} + b_{22}, E_2 | a_{11}, b_{11}) \\ &= Q(a_{11}, b_{11}) \cdot H(a_{11}, b_{11}) \\ &= M(a_{11}, b_{11}) \end{aligned}$$

Since $P(E_1, E_2, E_3, E_4 | a_{11}, b_{11}) = P(E_1, E_2, E_3, \neg E_4 | a_{11}, b_{11})$, this tells us the probability that 1,1 is a Stackelberg equilibrium and the social optimum is

$$P = 2 \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} M(a_{11}, b_{11}) f_a(a_{11}) da_{11} f_b(b_{11}) db_{11}$$

Since all 4 entries in a 2×2 game are equally likely to be the social optimum and be played, this gives us

$$P(S.O.) = 8 \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} M(a_{11}, b_{11}) f_a(a_{11}) da_{11} f_b(b_{11}) db_{11}$$

□

We see from Theorem 15 that although we have a method to exactly compute the probabilities of the social optimum being played in a zero-trust game, it is already quite complex for even a 2×2 game. Further, introducing the δ_i values for either the incomplete or complete knowledge policies vastly complicates the geometry of the space over which our probability distributions are computed. Therefore, assuming that f_a and f_b have finite expectation and variance, it will be generally more cost effective to use sampling methods to estimate the probability of playing the social optimum as well as related properties due to the relative ease of solving leader-follower games. Before we move on to Section 4.6 to do exactly that, we will first briefly discuss the expectation of zero trust leader-follower games, as a metric against which to measure the effectiveness of our policies.

Theorem 16. *The expected payoff of each player i in a $m_1 \times m_2 \times ... \times m_n$ zero-trust n-player Stackelberg game where each player i's payoffs for each entry in the probability tensor are generated iid from a distribution A_i is*

$$E[u_i|A_1, ..., A_n] = m_i \int_{-\infty}^{\infty} x f_{A_i}(x)(F_{A_i}(x))^{m_i - 1} dx$$

where f_{A_i} is the probability density function for A_i and F_{A_i} is the cumulative distribution function.

Proof. Consider player i's choice: player 1 makes its choice based on all other players' responses, however player 1's payoff is independent of these players. Players 2 through $i-1$ have also made choices based on all later players responses, but again each of their payoffs is independent of these later players. Thus, after players 1 through $i-1$ have made their choices, player i chooses between m_i possible responses, each with an iid payoff and each independent of the other $n-1$ players' payoffs despite the fact that previous players factored i's action into their choices, and i will factor later players into its choices. Player i will therefore be choosing the maximum of m_i iid variables and its payoff is distributed as the maximum of m_i samples from its payoff distribution A_i.

Let X be a random variable and let $X^{(k)}$ be the maximum of k iid samples of X. For its CDF we have $F_X^{(k)}(x) = (F_X(x))^k$ which gives us a PDF of $f_X^{(k)}(x) = k f_X(x)(F_X(x))^{k-1}$. Therefore, the expected value of $X^{(k)}$ is

$$E(X^{(k)}) = k \int_{-\infty}^{\infty} x f_X(x)(F_X(x))^{k-1} dx$$

which means the expected value of the game for player i is

$$E(u_i|A_1, ..., A_n) = E(A_i^{(m_i)}) = m_i \int_{-\infty}^{\infty} x f_{A_i}(x)(F_{A_i}(x))^{m_i - 1} dx.$$

□

Theorem 17. *The social optimum of a $m_1 \times m_2 \times ... \times m_n$ zero-trust n-player* *game with each player's payoffs generated iid from A_i is*

$$\mathcal{M}\int_{-\infty}^{\infty} x f_X(x)(F_X(x))^{\mathcal{M}-1}dx$$

where X is distributed as $\sum_{i=1}^{n} A_i$ for independent A_i and $\mathcal{M} = \prod_{i=1}^{n} m_i$.

Proof. The social optimum is defined as the maximum value of $\sum_{i=1}^{k} u_i(\sigma)$ across all strategies $\sigma \in \Sigma$, and is always a pure strategy profile. There are $\mathcal{M}$ such pure-strategy σ, all iid. By the same reasoning as in the proof of Theorem 16, this means the expected value of the social optimum is

$$\mathcal{M}\int_{-\infty}^{\infty} x f_X(x)(F_X(x))^{\mathcal{M}-1}dx$$

for $X \sim \sum_{i=1}^{n} A_i$. □

Given the results of Theorems 16 and 17, we can conclude that the expected PoA of a zero-trust 2-player $m \times n$ leader-follower maximization game is

$$\frac{E[(A+B)^{(mn)}]}{E[A^{(m)}] + E[B^{(n)}]}$$

where $X^{(n)}$ is the maximum of n iid samples of a distribution X.

APPENDIX B

SOCIAL NETWORK GAMES

B.1 Additional Notes: Rate of learning δ_{-i}

In Section 3.3 we looked at how a player i can learn δ_j through interactions with player j as both a leader and a follower. In this appendix we provide bounds on how quickly i can learn a fixed δ_j. We focus on doing so from i's perspective as a leader, rather than as a follower: although we saw in Section 3.3.2 that information about δ_j can be inferred when j is the leader, i cannot guarantee that j will ever invite it to interact. Further, the likelihood of gaining this information changes with δ_{ij}, player j's estimate of δ_i. We will assume that j believes that $u_j(\theta_{ij}) \geq 0$, as otherwise it will never accept an invitation from i to interact. We now consider how many games are needed for i to determine δ_j to within an error of $\varepsilon > 0$.

First, we note that in order to guarantee that $|\delta_j - \delta_{ji}| \leq \varepsilon$ where δ_{ji} is player i's estimate of δ_j, both of the following must be true:

1. $\delta_j - \delta_{ji} \leq \varepsilon$
2. $\delta_{ji} - \delta_j \leq \varepsilon$

While trivial, this implies that if player i can determine an interval $[\delta^l_{ji}, \delta^u_{ji})$ such that $\delta_j \in [\delta^l_{ji}, \delta^u_{ji})$ and $\delta^u_{ji} - \delta^l_{ji} < 2\varepsilon$, then δ_j must be within ε of at least one of the lower bound δ^l_{ji} or the upper bound δ^u_{ji}. Consider the expected number of observations required for player i to determine both an upper and lower bound within ε of δ_j. As this is a stricter condition than is required to estimate δ_j to within an error of ε, determining it serves as an upper bound on the expected number of observations to estimate δ_j to within ε.

Suppose that player 1 and player 2 are participating in a randomly generated $m \times n$ leader-follower game generated via $\mathcal{A}_{12}, \mathcal{B}_{12}$ in which all entries of each matrix are generated independently and identically to all other entries in the matrix. Without loss of generality, we will assume that player 1 chooses to play s_i as the leader. We want to determine the probability that the follower player 2 will choose a strategy s_j which reveals a lower bound δ_{21}^l such that $\delta_2 - \delta_{21}^l \leq \varepsilon$. Note that a leader-follower game after the leader has chosen its strategy is equivalent to a $1 \times n$ game, as in Table 3.4 in Section 3.3.1. In order for player 1 to observe a lower bound $\delta_{21}^l > 0$, it is necessary for player 2 to not pick the s_j in Table 3.4 that maximizes its utility. Without loss of generality, assume $b_1 \geq b_j \ \forall i \in [n]$, where $[n] = \{1, 2, ..., n\}$. If player 2 instead chooses to play s_j, then player 1 can deduce that $\delta_2 \geq b_1 - b_j$ and set $\delta_{21}^l = b_1 - b_j$ using Algorithm 1. Based on the fact that all such values are randomly generated from known distributions $\mathcal{A}_{12}, \mathcal{B}_{12}$, we can compute the probability of finding an acceptable lower bound δ_{21}^l by taking the cumulative distribution functions of the distributions over the relevant areas in $\mathcal{R}^2$.

Lemma 9. *For the game in Table 3.4, the probability of the game revealing δ_{21}^l such that $\delta_2 - \delta_{21}^l \leq \varepsilon$ is*

$$P^l(\varepsilon) = n(n-1) \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} \left(\int_{b_1-\delta_2}^{b_1-\delta_2+\varepsilon} \left(\int_{a_1+b_1-b_2}^{\infty} P(E|s_1, s_2) f_a(a_2) da_2 \right) f_b(b_2) db_2 \right) f_a(a_1) da_1 \right) f_b(b_1) db_1$$

where

$$P(E|s_1, s_2) = \left(\int_{b_1-\delta}^{b_1} \left(\int_{\infty}^{a_2+b_2-b_j} f(a_j) da_j \right) f(b_j) db_j + \int_{-\infty}^{\infty} \left(\int_{-\infty}^{b_1-\delta_2} f_b(b_j) db_j \right) f_a(a_j) da_j \right)^{n-2}.$$

Proof. We begin by considering the probability of an event E occurring where E is the event that b_1 is the greedy choice for player 2, s_2 is chosen, and this results in a lower bound δ_{21}^l within ε of the true δ_2. In order for this to occur, the relationship between s_1 and s_2 must be as in Figure B.1, where both s_2 gives better net utility than s_1 and b_2 is within ε of δ_2 but not greater than it. The diagonal line which passes through s_1 is the set of all points which provide equal net utility to s_1: points above the line provide better net utility and points below it provide worse net utility. Additionally, given s_1 and s_2 it is necessary for another strategy s_j where $j \neq 1, 2$ to not both give better net utility than s_2 and have $b_1 - b_j \leq \delta_2$. If both of these conditions are met, s_j will be picked instead of s_2. Therefore, for a given s_1, s_2 that fulfill the necessary conditions, s_j must lie in the shaded region of Figure B.2 for event

E to occur. Thus if E_j is the event that s_j is in an acceptable position

$$P(E_j|s_1, s_2) = \int_{b_1-\delta}^{b_1} \left(\int_{\infty}^{a_2+b_2-b_j} f(a_j)da_j \right) f(b_j)db_j + \int_{-\infty}^{\infty} \left(\int_{-\infty}^{b_1-\delta_2} f_b(b_j)db_j \right) f_a(a_j)da_j.$$

This must be true for every s_j with $j \neq 1, 2$, so $P(E|s_1, s_2) = P(E_j|s_1, s_2)^{n-2}$. Next we note that for event E to occur for a given s_1, we require that s_2 lie in the shaded area in Figure B.1 which occurs with probability

$$P(E|s_1) = \int_{b_1-\delta_2}^{b_1-\delta_2+\varepsilon} \left(\int_{a_1+b_1-b_2}^{\infty} P(E|s_1, s_2) f_a(a_2)da_2 \right) f_b(b_2)db_2.$$

This allows us to integrate over all values of s_1 to get that

$$P(E) = \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} P(E|s_1) f_a(a_1)da_1 \right) f_b(b_1)db_1.$$

Given that any s_i could be the greedy response and any s_j for $j \neq i$ could provide the bound upper bound with equal probability, but each of these events is mutually exclusive, we finally get that $P^l(\varepsilon) = n(n-1)P(E)$, which completes the proof. □

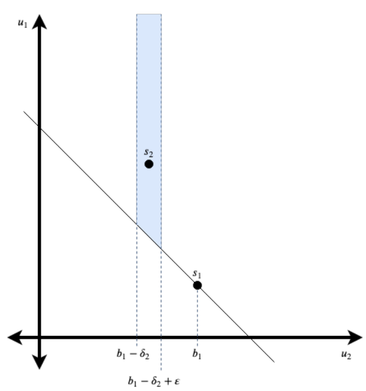

Figure B.1: Area for s_2 given s_1 to establish lower bound

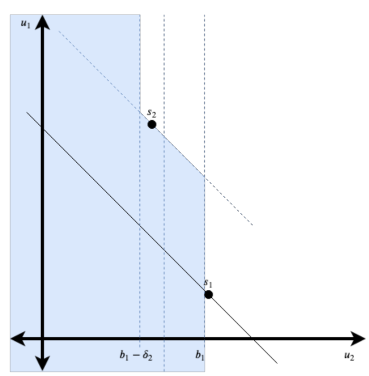

Figure B.2: Area for s_i given s_1, s_2 to establish lower bound

The role of n as indicated by Lemma 9 is somewhat counterintuitive. The Lemma shows that the probability of the first player establishing an acceptable lower bound δ_{21}^l on δ_2 goes to 0 as the second player's number of strategies $n \to \infty$: if the second player is presented with more choices, shouldn't they have a higher probability of getting one which allows them to precisely play near δ_2 and maximize utility? This may be the case in smaller values of

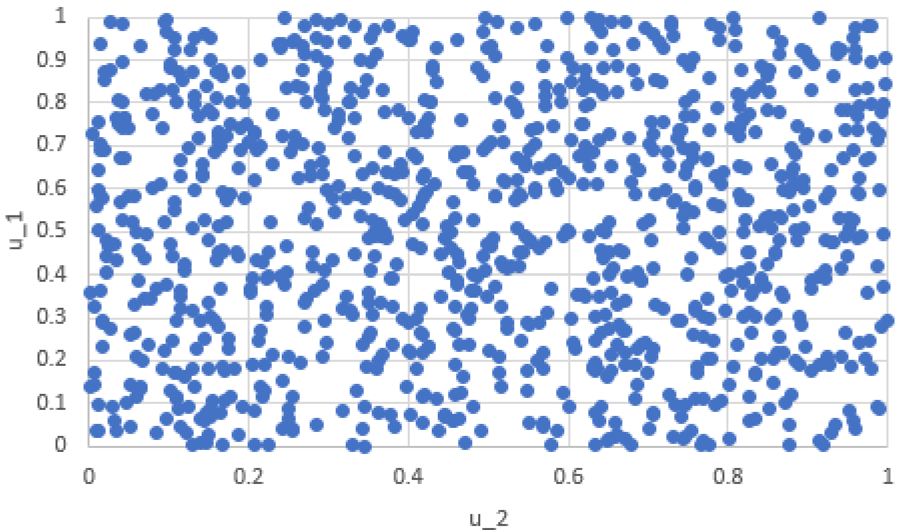

Figure B.3: Example game from Table 3.4, $n = 1000$ $f_a = f_b = U[0, 1]$

n, but it does not occur in general.

Suppose $f_a = f_b = U[0, 1]$, the uniform distribution. For high values of n, when the first player chooses strategy s_i the second player will have a greedy best response $s_j = G_2(s_i)$ with $b_{ij} \approx 1$ with high probability, as for a high number of independent samples of $U[0, 1]$, the expected value of the maximum sample goes to 1. Also due to the high value of n, there will be another response s_l with $b_{il} < b_{ij}$ but $b_{il} \approx a_{il} \approx 1$ with high probability due to the same independent sampling. Consider this game with $n = 1000$ and suppose the first player selects strategy s_i. The second player's strategy set now resembles Figure B.3, which is a randomly generated row of the $m \times 1000$ game matrix. In the figure, a δ_2 of approximately 0.02 appears to be sufficient for the socially optimal strategy near the top right corner to be played. This means that if, for example, $\delta_2 = 0.2$ it will be nearly impossible to ever establish an acceptable lower bound within $\varepsilon = 0.05$ of δ_2; it is similarly difficult to establish any upper bound at all.

Next we derive the probability player 1 observes an upper bound $\delta_{21}^u < \infty$ on δ_2 in the game in Table 3.4. For this to occur, player 2 must choose s_i over s_j because $b_i > b_j$ despite the fact that $a_i + b_i < a_j + b_j$. If, without loss of generality, s_1 is the greedy best response for player 2, this allows player 1 to conclude that $\delta_2 < b_1 - b_j$ and set $\delta_{21}^u = b_1 - b_j$. As we have already seen, it would also allow player 1 to set $\delta_{21}^l = b_1 - b_j$, thus showing that both upper and lower bounds can be observed within a single game.

Lemma 10. *For the game in Table 3.4, the probability of the game revealing δ_{21}^u such that $\delta_{21}^l - \delta_2 \leq \varepsilon$ for $\varepsilon > 0$ is*

$$P^u(\varepsilon) = n \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} \left(P(E \cap E_1|s_1) + (n-1)P(E \cap E_2|s_1)\right) f_a(a_1) da_1 \right) f_b(b_1) db_1$$

where

$$P(E \cap E_1|s_1) = P(s_i \in A_1|s_1)^{n-1} - (P(s_i \in A_1|s_1) - P(s_i \in A_2|s_1))^{n-1},$$

$$P(s_i \in A_1|s_1) = \int_{-\infty}^{b_1-\delta_2} \left(\int_{-\infty}^{\infty} f_a(a_i) da_i \right) f_b(b_i) db_i + \int_{b_1-\delta_2}^{b_1} \left(\int_{-\infty}^{a_1+b_1-b_i} f_a(a_i) da_i \right) f_b(b_i) db_i,$$

$$P(s_i \in A_2|s_1) = \int_{b_1-\delta_2-\varepsilon}^{b_1-\delta_2} \left(\int_{a_1+b_1-b_i}^{\infty} f_a(a_i) da_i \right) f_b(b_i) db_i,$$

and

$$P(E \cap E_2|s_1) = \int_{b_1-\delta_2}^{b_1} \left(\int_{a_1+b_1-b_2}^{\infty} \left(P(A_3|s_1, s_2)^{n-2} - (P(A_3|s_1, s_2) - P(A_4|s_1, s_2))^{n-2}\right) f_a(a_2) da_2 \right) f_b(b_2) db_2,$$

$$P(s_i \in A_3|s_1, s_2) = \int_{b_1-\delta}^{b_1} \left(\int_{\infty}^{a_2+b_2-b_i} f(a_i) da_i \right) f b_i db_i + \int_{-\infty}^{\infty} \left(\int_{-\infty}^{b_1-\delta_2} f_b(b_i) db_i \right) f_a(a_i) da_i,$$

$$P(s_i \in A_4|s_1, s_2) = \int_{b_1-\delta_2-\varepsilon}^{b_1-\delta_2} \left(\int_{a_2+b_2-b_i}^{\infty} f_a(a_i) da_i \right) f_b(b_i) db_i.$$

Proof. Despite its intimidating look this lemma is simply the result of integrating probability distribution functions over $\mathcal{R}^2$. Consider the event E that s_1 is the greedy choice for player 2. There are two possible outcomes: player 2 chooses s_1 (event E_1) or player 2 chooses another strategy (without loss of generality we assume that strategy to be s_2) with better net utility such that $b_1 - b_2 \leq \delta_2$ (event E_2). We refer to these two outcomes as (respectively) case 1 and case 2.

Case 1 occurs if all strategies s_i are in the shaded region in Figure B.4. As a function of s_1, a strategy s_i is in this region A_1 with probability

$$P(s_i \in A_1|s_1) = \int_{-\infty}^{b_1-\delta_2} \left(\int_{-\infty}^{\infty} f_a(a_i) da_i \right) f_b(b_i) db_i + \int_{b_1-\delta_2}^{b_1} \left(\int_{-\infty}^{a_1+b_1-b_i} f_a(a_i) da_i \right) f_b(b_i) db_i.$$

Next, for fixed s_1, we need to consider the probability that case 1 occurs and at least one strategy is able to provide an upper bound on δ_2. This occurs if all strategies lie within the shaded area in Figure B.4 and at least one strategy lies in the shaded area A_2 in Figure B.5. The probability of a strategy lying in A_2 is

$$P(s_i \in A_2|s_1) = \int_{b_1-\delta_2-\varepsilon}^{b_1-\delta_2} \left(\int_{a_1+b_1-b_i}^{\infty} f_a(a_i)da_i \right) f_b(b_i)db_i,$$

which means that case 1 occurs and an acceptable bound is established with probability $P(E \cap E_1|s_1) = P(s_i \in A_1|s_1)^{n-1} - (P(s_i \in A_1|s_1) - P(s_i \in A_2|s_1))^{n-1}$.

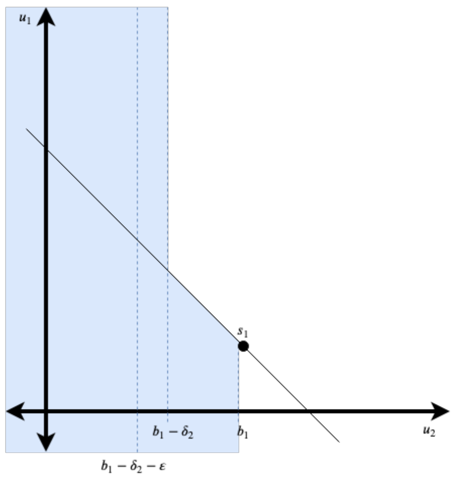

Figure B.4: Area for s_2 given s_1 to establish upper bound, case 1

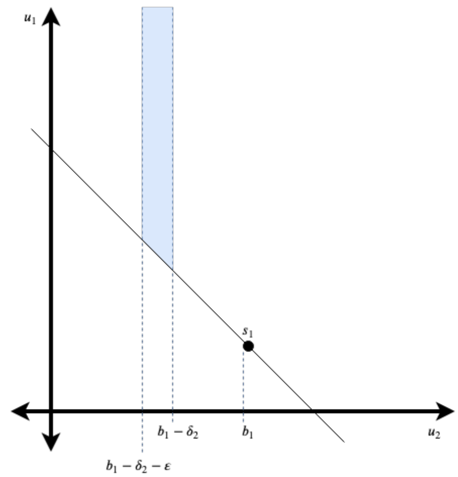

Figure B.5: Area for s_i given s_1 to establish upper bound, case 1

We now consider case 2, that another strategy is played, and assume without loss of generality that the strategy played is s_2. Case 2 requires that occurs if all strategies s_i are in the shaded region in Figure B.6, for $i \neq 1, 2$. As a function of s_1 and s_2, a strategy s_i is in this region with probability

$$P(A_2|s_1, s_2) = \int_{-\infty}^{b_1-\delta_2} \left(\int_{-\infty}^{\infty} f_a(a_i)da_i \right) f_b(b_i)db_i + \int_{b_1-\delta_2}^{b_1} \left(\int_{-\infty}^{a_2+b_2-b_i} f_a(a_i)da_i \right) f_b(b_i)db_i.$$

Next, we want to consider the probability that case 2 occurs and an acceptable upper bound is observed. For a fixed s_1, s_2, s_2 is played if all other s_i lie in the shaded region A_3 in Figure B.2, the probability of which we know from the proof of Lemma 9 is

$$P(s_i \in A_3|s_1, s_2) = \int_{b_1-\delta}^{b_1} \left(\int_{\infty}^{a_2+b_2-b_i} f(a_i)da_i \right) fb_idb_i + \int_{-\infty}^{\infty} \left(\int_{-\infty}^{b_1-\delta_2} f_b(b_i)db_i \right) f_a(a_i)da_i.$$

In order for one of these s_i to provide an acceptable upper bound, at least one of them must be in the shaded region A_4 in Figure B.6, providing better

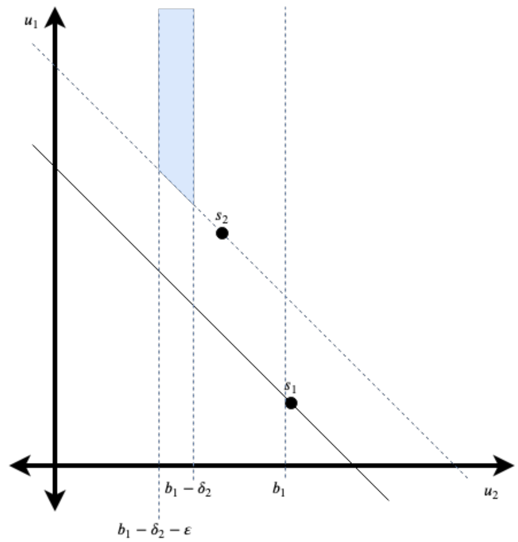

Figure B.6: Area for s_i given s_1, s_2 to establish upper bound, case 2

net utility than s_2 and providing utility b_i such that $b_1 - \delta_2 - \varepsilon \le b_i \le b_1 - \delta_2$. The probability of an s_i being in this region is

$$P(s_i \in A_4 | s_1, s_2) = \int_{b_1-\delta_2-\varepsilon}^{b_1-\delta_2} \left(\int_{a_2+b_2-b_i}^{\infty} f_a(a_i) da_i \right) f_b(b_i) db_i,$$

which means that case 2 occurs and establishes an acceptable upper bound with probability $P(A_3|s_1, s_2)^{n-2} - (P(A_3|s_1, s_2) - P(A_4|s_1, s_2))^{n-2}$. Note that this value is zero for $n = 2$, indicating that in a two strategy game an upper bound cannot be established if case 2 occurs. Therefore, as a function of s_1 the probability case 2 occurs and an acceptable upper bound is established is

$$P(E \cap E_2|s_1) = \int_{b_1-\delta_2}^{b_1} \left(\int_{a_1+b_1-b_2}^{\infty} \left(P(A_3|s_1, s_2)^{n-2} - (P(A_3|s_1, s_2) - P(A_4|s_1, s_2))^{n-2} \right) f_a(a_2) da_2 \right) f_b(b_2) db_2.$$

Finally, this gives us that the probability of establishing an acceptable upper bound is

$$P^u(\varepsilon) = n \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} (P(E \cap E_1|s_1) + (n-1) P(E \cap E_2|s_1)) f_a(a_1) da_1 \right) f_b(b_1) db_1.$$

□

As we noted before the lemma, when $n \ge 3$, it is possible for both upper and lower bounds to be established in a single game. The derivation of

Lemma 10 allows us to do so directly through our derivation of $P(E \cap E_2|s_1)$, which was the probability that the player 2 did not select the greedy best response but still revealed an upper bound within ε of δ_2.

Lemma 11. *The probability of player 1 observing both an upper and lower bound within ε of δ_2 is given by*

$$Q(\varepsilon) = n(n-1) \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} (P'(E \cap E_2|s_1)) f_a(a_1) da_1 \right) f_b(b_1) db_1.$$

where

$$P'(E \cap E_2|s_1) = \int_{b_1-\delta_2}^{b_1-\delta_2+\varepsilon} \left(\int_{a_1+b_1-b_2}^{\infty} \left(P(A_3|s_1,s_2)^{n-2} - (P(A_3|s_1,s_2) - P(A_4|s_1,s_2))^{n-2}\right) f_a(a_2) da_2 \right) f_b(b_2) db_2.$$

Theorem 18. *For the game in Table 3.4 the expected number of games for player 1 to get an estimate δ_{21} of δ_2 guaranteed to have error most ε from the true value is less than or equal to*

$$E[\mathcal{T}(\varepsilon)] \leq T(\varepsilon) = \frac{1}{P^u(\varepsilon)+P^l(\varepsilon)-Q(\varepsilon)} \left(1 + \frac{P^u(\varepsilon)-Q(\varepsilon)}{P^l(\varepsilon)} + \frac{P^l(\varepsilon)-Q(\varepsilon)}{P^u(\varepsilon)}\right).$$

Proof. We note that the only way for player 1 to make an estimate of δ_2 which is guaranteed to be within at most ε of the true value is to find an interval $[\delta_{21}^l, \delta_{21}^u)$ such that $\delta_{21}^u - \delta_{21}^l \leq 2\varepsilon$. Next, we note that in order to obtain this interval at least one of δ_{21}^l and δ_{21}^u must be within ε of δ_2. Therefore we can find an upper bound on the expected number of games required by finding the expected number of games required to observe both upper and lower bounds within ε of δ_2.

Beginning from the first game, the expected time to discover one or more bounds is $\frac{1}{P^u(\varepsilon)+P^l(\varepsilon)-Q(\varepsilon)}$. This event can occur in any of three ways: an acceptable upper bound is found, an acceptable lower bound is found, or both are found. The probabilities of these events are proportional to $P^u(\varepsilon)-Q(\varepsilon)$, $P^l(\varepsilon)-Q(\varepsilon)$, and $Q(\varepsilon)$ respectively. The expected total time to discover both is therefore

$$\frac{1}{P^u(\varepsilon)+P^l(\varepsilon)-Q(\varepsilon)} + \frac{P^u(\varepsilon)-Q(\varepsilon)}{P^u(\varepsilon)+P^l(\varepsilon)-Q(\varepsilon)} \frac{1}{P^l(\varepsilon)} + \frac{P^l(\varepsilon)-Q(\varepsilon)}{P^u(\varepsilon)+P^l(\varepsilon)-Q(\varepsilon)} \frac{1}{P^u(\varepsilon)}$$

$$= \frac{1}{P^u(\varepsilon)+P^l(\varepsilon)-Q(\varepsilon)} \left(1 + \frac{P^u(\varepsilon)-Q(\varepsilon)}{P^l(\varepsilon)} + \frac{P^l(\varepsilon)-Q(\varepsilon)}{P^u(\varepsilon)}\right).$$

□

With the completion of Theorem 18, we now have an upper bound on the expected time to discover both an upper and lower bound within ε of the game in Table 3.4. However, we are interested in that $1 \times n$ game because it is equivalent to an $m \times n$ game in which the leader has made its decision and is waiting for the follower. Now we return to our original goal, estimating the expected time for the leader in an $m \times n$ game to estimate the follower's δ value to within ε.

If player 1's goal is to learn δ_2, rather than to play according to δ_1, the challenge it faces is deciding which of its m strategies to select each game. Note also that in pursuing this behavior player 1 has decided to focus purely on exploration and has abandoned any interest in its own utility, which means that it is impossible for its neighbors to learn anything about δ_1 based on its actions as a leader. Theorem 18 implies that the expected time for strategies which reveal an upper and lower bound within ε of δ_2 to occur in a randomly generated game is $\frac{1}{m}T(\varepsilon)$, which provides an upper bound on the expected time of player 1's theoretical optimum strategy. Similarly, it implies that if the leader chooses its strategy s_i randomly, it has an upper bound of $T(\varepsilon)$ on the expected time to achieve this estimate. However, choosing in such a random manner ignores what the leader has already learned from the follower: while it may not have $\delta_{21}^u - \delta_{21}^l \leq 2\varepsilon$, it will still over time gain some $[\delta_{21}^l, \delta_{21}^u)$ interval in which δ_2 is located through the use of Algorithm 1. This allows the leader to determine whether or not it will refine its knowledge of δ_2 by selecting s_i. Assume without loss of generality that s_1 is the follower's greedy best response to the leader selecting s_i: if the Pareto frontier of player 2's responses to s_i contains an s_j such that $\delta_{21}^l < b_1 - b_j < \delta_{21}^u$, selecting s_i will result in the player 1 refining at least one of the bounds on δ_2. If the frontier does not contain such an s_j, there will be no improvement in the bounds by selecting s_i. Figure 3.2 in Section 3.3.1 gives an example of this with player 2's potential response s_3: if player 2 responds with s_3 then the lower bound δ_{21}^l will be raised, and if it responds with s_2 the upper bound δ_{21}^u will be lowered.

B.2 Derivations

B.2.1 Proof of Theorem 8

Theorem 8 *Consider a social network G with uniform interactions $\mathcal{A}_{ij} = \mathcal{B}_{lh}$ for all $l, i, j, h \in [N]$ such that all payoffs are nonnegative and for agent i with neighbors j and l, $\delta_j \leq \delta_l \to u_i(\theta_{ij}) \leq u_i(\theta_{il})$. Then the N-player metagame with closed interval strategy space $\Delta_i \subseteq \mathcal{R}$ and utility function $\mathbf{u}_i$ for $i \in [N]$ possesses a mixed Nash equilibrium.*

For simplicity of notation, in the this proof we will use $\mathbf{u}_i(\delta_i, \delta_{-i})$ instead of $\mathbf{u}_i(\theta_{-i}, \delta_i)$. Also, before formally starting the proof we first state the following result:

Theorem 19 ([57]). *Let $\Sigma_i \subseteq \mathcal{R}$ for $i \in [N]$ be a closed interval and let $U_i : \Sigma \to \mathcal{R}$ be continuous except on a subset $\Sigma^{**}(i) \subseteq \Sigma^*(i)$. If $\sum_{i=1}^{N} U_i(\sigma)$ is upper semi-continuous and $U_i(\sigma_i, \sigma_{-i})$ is bounded and weakly lower semi-continuous in σ_i then the N-player game with closed interval strategy space $\Sigma_i \subseteq \mathcal{R}$ and utility function U_i for $i \in [N]$ possesses a mixed-strategy Nash equilibrium.*

[57] uses the following definition of weak lower semi-continuity in $\mathcal{R}$:

Definition 13 (Weak lower semi-continuity). *$U_i(\sigma_i, \sigma_{-i})$ is weakly lower semi-continuous in σ_i if $\forall \sigma_i' \in \Sigma_i^{**}(i)$, $\exists \lambda \in [0, 1]$ such that $\forall \sigma_{-i} \in \Sigma_{-i}^{**}(\sigma_i')$,*

$$\lambda \lim \inf_{\sigma_i \xrightarrow{-} \sigma_i'} U_i(\sigma_i, \sigma_{-i}) + (1 - \lambda) \lim \inf_{\sigma_i \xrightarrow{+} \sigma_i'} U_i(\sigma_i, \sigma_{-i}) \geq U_i(\sigma_i', \sigma_{-i})$$

Proof. Proof of Theorem 8: This proof will make use of Theorem 19. As such, we need to show three things: a set $\Delta^*(i) = \{(\delta_1, ..., \delta_N) \in \Delta | \exists j \neq i, \exists d, 1 \leq d \leq D(i) \text{ such that } \delta_j = f_{ij}^d(\delta_i)\}$ which appropriately captures discontinuities in $\mathbf{u}_i(\delta)$, that $\mathbf{u}_i(\delta_i, \delta_{-i})$ is bounded and weakly lower semi-continuous in δ_i, and that $\sum_{i \in [N]} \mathbf{u}_i(\delta)$ is upper semi-continuous.

We begin by determining the set $\Delta^*(i)$. We earlier noted that $\mathbf{u}_i(\delta_i, \delta_{-i})$ has at most $|N_i^1|$ discontinuities in δ_i when all interactions are uniform and nonnegative and $\delta_j \leq \delta_l \to u_i(\theta_{ij}) \leq u_i(\theta_{il})$, and that all of them occur in $w_i(\delta_i, \delta_{-i})$. w_i is the utility gained by agent i receiving invitations. Therefore, if i receives an invitation from agent j when δ_i changes, another agent l that previously received an invitation from j now loses it. Given that $\delta_i \leq \delta_l \rightarrow u_j(\theta_{ji}) \leq u_j(\theta_{jl})$, this discontinuity occurs when $\delta_i = \delta_l$. Therefore, we can let $D(i) = 1$, $f^d_{ij}(x) = x$, the identity function, and

$$\Delta^*(i) = \{(\delta_1, ..., \delta_N) \in \Delta | \exists j \neq i, \text{ such that } \delta_j = \delta_i\}$$

will contain all potential discontinuities.

Next we will show that $\mathbf{u}_i(\delta_i, \delta_{-i})$ is bounded and weakly lower semi-continuous in δ_i. From Theorem 7 we observe that $u_i(\theta_{ji})$ and $u_i(\theta_{ij})$ are both continuous in δ_i. Therefore all discontinuities in $w_i(\delta_i, \delta_{-i})$ occur due to K^2_i changing. Consider one such discontinuity point δ': We know that there is an agent l such that $\delta'_i = \delta'_l$ and there is another agent j which i and l both neighbor who is now indifferent between sending an invitation to agent i and agent l. Let K^2_i be the set of invitations i receives from other agents $h \neq j$.

$$\begin{aligned}
\liminf_{\delta_i \xrightarrow{-} \delta'_i} \mathbf{u}_i(\delta_i, \delta'_{-i}) &= \liminf_{\delta_i \xrightarrow{-} \delta'_i} v_i(\delta_i, \delta'_{-i}) + \sum_{h \in K^2_i} u_i(\delta'_h, \delta_i) \\
&= v_i(\delta'_i, \delta'_{-i}) + \sum_{h \in K^2_i} u_i(\delta'_h, \delta'_i) \\
\liminf_{\delta_i \xrightarrow{+} \delta'_i} \mathbf{u}_i(\delta_i, \delta'_{-i}) &= \liminf_{\delta_i \xrightarrow{+} \delta'_i} v_i(\delta_i, \delta'_{-i}) + \sum_{h \in K^2_i} u_i(\delta'_h, \delta_i) + u_i(\delta'_j, \delta_i) \\
&= v_i(\delta'_i, \delta'_{-i}) + \sum_{h \in K^2_i} u_i(\delta'_h, \delta'_i) + u_i(\delta'_j, \delta'_i).
\end{aligned}$$

Therefore $\mathbf{u}_i(\delta_i, \delta_{-i})$ is weakly lower semi-continuous by selecting $\lambda = 1$.

Finally, we prove $\mathbf{u}(\delta) = \sum_{i \in [N]} \mathbf{u}_i(\delta)$ is upper semi-continuous. Actually, we will prove the stronger condition that it is continuous. Let δ' be a point of discontinuity for some $\mathbf{u}_i$. As we have discussed, this occurs due to some other agent j shifting on whether or not to issue an invitation to agent i or another of its neighbors agent l (the case where it is actually a set of neighbors L follows naturally). As a consequence, δ' is also a point of discontinuity for $\mathbf{u}_l$. However, it is not a point of discontinuity for $\mathbf{u}_j$. Theorem 7 shows $u_j(\theta_{ji})$ and $u_j(\theta_{jl})$ are continuous in δ_i and δ_l, respectively. Therefore $w_j(\delta)$ is continuous in both. While the set of invites j issues, K^1_j, is subject to change, $v_j(\delta)$ the sum of the k_j highest values in the set of functions $\{u_j(\theta_{jh})\}_{h \in N^1_j}$, all of which are continuous in δ and is therefore continuous as well. This means $\mathbf{u}_j(\delta)$ is continuous, leaving us to show that while $\mathbf{u}_i$ and $\mathbf{u}_l$ are discontinuous at δ', the sum of the two functions is not. Again, we only need concern ourselves with showing $w_i + w_l$ is continuous.

$$\begin{aligned}
\lim_{\delta_i \xrightarrow{-} \delta_i'} w_i(\delta_i, \delta'_{-i}) + w_l(\delta_i, \delta'_{-i}) &= \lim_{\delta_i \xrightarrow{-} \delta_i'} w_i(\delta_i, \delta'_{-i}) + \lim_{\delta_i \xrightarrow{-} \delta_i'} w_l(\delta_i, \delta'_{-i}) \\
&= \sum_{h \in K_i^1 \setminus j} u_i(\delta'_h, \delta_i) + \sum_{h \in K_l^1 \setminus j} u_l(\delta'_h, \delta'_l) + u_l(\delta'_j, \delta'_l) \\
\lim_{\delta_i \xrightarrow{+} \delta_i'} w_i(\delta_i, \delta'_{-i}) + w_l(\delta_i, \delta'_{-i}) &= \lim_{\delta_i \xrightarrow{+} \delta_i'} w_i(\delta_i, \delta'_{-i}) + \lim_{\delta_i \xrightarrow{+} \delta_i'} w_l(\delta_i, \delta'_{-i}) \\
&= \sum_{h \in K_i^1 \setminus j} u_i(\delta'_h, \delta_i) + \sum_{h \in K_l^1 \setminus j} u_l(\delta'_h, \delta'_l) + u_i(\delta'_j, \delta_i)
\end{aligned}$$

The fact that $u_l(\delta'_j, \delta'_l) = \lim_{\delta_i \to \delta'_i} u_i(\delta'_j, \delta_i)$ implies that the left- and right-side limits are equal, and that they are equal to the actual value

$$w_i(\delta'_i, \delta'_{-i}) + w_l(\delta'_i, \delta'_{-i}) = \textstyle\sum_{h \in K_i^1 \setminus j} u_i(\delta'_h, \delta'_i) + \sum_{h \in K_l^1 \setminus j} u_l(\delta'_h, \delta'_l) + \frac{1}{2} u_i(\delta'_j, \delta'_i) + \frac{1}{2} u_l(\delta'_j, \delta'_l)$$

at δ'. This implies that $\mathbf{u}_i + \mathbf{u}_l$ is continuous at δ' and hence that $u(\delta)$ is continuous. Therefore, the metagame of selecting $\delta_i \in \Delta_i$ possesses a mixed Nash equilibrium in this setting.

$\square$

B.3 Additional Figure

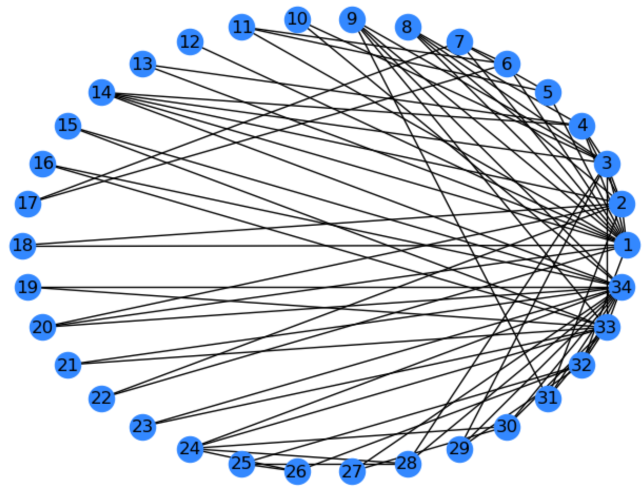

Figure B.7: Zachary's Karate Club

APPENDIX C

PRIZE COLLECTING MULTI-AGENT ORIENTEERING

C.1 Integer Program Formulations, Algorithms, and Numerical Results

$$\text{IP1:}\ \max \sum_{i=1}^{n} c_i z_i$$

subject to

$$\sum_{i|(j,i)\in E} x_{ji}^m = \sum_{i|(i,j)\in E} x_{ij}^m + d_j \qquad \forall j \in [n], m \in [k], \tag{C.1}$$

$$z_j \leq \sum_{m=1}^{k} \sum_{i|(i,j)\in E} x_{ij}^m \qquad \forall j \in [n], \tag{C.2}$$

$$r \geq \sum_{(i,j)\in E} l_{ij} x_{ij}^m \qquad \forall m \in [k], \tag{C.3}$$

$$x_{ij}^m, z_j \in \{0,1\} \qquad \forall i, j \in [n], m \in [k]. \tag{C.4}$$

$$\text{IP2:}\ \min \sum_{i=1}^{n} c_i z_i^2$$

subject to

$$\sum_{i|(j,i)\in E} x_{ji} = \sum_{i|(i,j)\in E} x_{ij} + d_j \qquad \forall j \in [n], \tag{C.5}$$

$$x_{ij} \le a_{ij} \qquad \forall i, j \in [n], \tag{C.6}$$

$$v_j^2 \le \sum_{i|(i,j)\in E} x_{ij} \qquad \forall j \in [n], \tag{C.7}$$

$$v_j^2 \ge \frac{1}{n} \sum_{i|(i,j)\in E} x_{ij} \qquad \forall j \in [n], \tag{C.8}$$

$$r \ge \sum_{(i,j)\in E} l_{ij} x_{ij}, \tag{C.9}$$

$$z_i^1 + z_i^2 \le 1 \qquad \forall j \in [n], \tag{C.10}$$

$$z_i^1 \ge (V_i^1 * t_i^2 - T_i^1)/M \qquad \forall i \in [n], \tag{C.11}$$

$$z_i^2 \ge (v_i^2 * T_i^1 - t_i^2)/M \qquad \forall j \in [n], \tag{C.12}$$

$$t_j^2 \ge t_i^2 + (x_{ij} - 1)M + l_{ij} x_{ij} \qquad \forall i, j \in [n], \tag{C.13}$$

$$x_{ij}, z_j^m, v_j^2 \in \{0, 1\} \qquad \forall i, j \in [n], \qquad m \in [k]. \tag{C.14}$$

Algorithm 3 Algorithm 1: ReservedSolution(G, k, prizes)

Require: G,k,prizes
  Initialize $sum \leftarrow 0$;
  Initialize $Paths$ as a size k array;
  for $i = 1$ to k **do**
    $path$, $score \leftarrow$ SolveIP1($G, k \leftarrow 1$,prizes);
    for j in $path$ **do**
      prizes[j] $\leftarrow 0$;
    end for
    $sum \leftarrow sum + score$, $Paths[i] \leftarrow path$;
  end for
  return $Paths$, sum

Algorithm 4 UnreservedSolution(G, prizes)

Initialize $score1 \leftarrow 0$, $score2 \leftarrow 0$, $path1$, $path2$;
Initialize $model \leftarrow$ TOP($G, k = 1$,prizes)
while $True$ **do**
    $p1$, $s1 \leftarrow$ SolveIP1($model$);
    if $s1 \leq score1$ **then**
        return $score1, path1, score2, path2$
    else
        $p2 \leftarrow$ SolveIP2($G, p1$,prizes);
        $s1$, $s2 \leftarrow$ ComputeScores($G, p1, p2$,prizes);
        if $s1 > score1$ **then**
            $score1 \leftarrow s1$, $score2 \leftarrow s2$, $path1 \leftarrow p1$, $path2 \leftarrow p2$;
        end if
        $model$.AddConstraint($p1$ is infeasible)
    end if
end while

Algorithm 5 TurnSolution(G, prizes)

Initialize hashset Space $\leftarrow \varnothing$;
Initialize nCur $\leftarrow$ 1, nNext $\leftarrow$ 1, rCur $\leftarrow r$, rNext $\leftarrow r$, colBetween $\leftarrow \varnothing$;
Initialize state0 $\leftarrow$ (nCur, nNext, rCur, rNext, colBetween);
run SolveState(state0)
return ConstructPaths(Space)

Algorithm 6 SolveState(state)

```
n1 ← state.nCur, n2 ← state.nNext, r1 ← state.rCur, r2 ← state.rNext;
curPrize ← prizes, collected ← state.collectedBetween;
for node in collected do
    curPrize[node] = 0;
end for
if state in Space then return
else if n2 = n then
    path, score = SolveIP1(G,curPrize,startFrom←n1, r1);
    if IP1 is infeasible then
        valCur ← −∞;
    else valCur ← score;
    end if
    valNext ← 0, nextState ← (n, n, 0, 0, ∅);
    nextStep ← path;
    Space[state] ← (valCur, valNext, nextState, nextStep);
    return
else
    Space[state] ← (−∞, −∞, ∅, n);
    nextMoves ← possible moves for current player;
    for j in nextMoves do
        collectedBetween ← ∅;
        for i ∈ [min (node2, j), max (node2, j)] do
            if i ∈ collected then
                collectedBetween.append(i)
            end if
        end for
        newState ← (n2,j,r2,r1-1,collectedBetween);
        SolveState(newState);
        valNext ← StateSpace[newState].valNext;
        valCur ← StateSpace[state].valCur;
        p ← curPrize[j] +valNext;
        if p > valCur then
            newValN ← StateSpace[newState].valCur;
            Space[state] ← (p,newValN,newState,j);
        end if
    end for
end if
```

REFERENCES

[1] J. F. Nash, “Equilibrium points in n-person games,” *Proceedings of the National Academy of Sciences*, vol. 36, no. 1, pp. 48–49, 1950.

[2] A. Vetta, “Nash equilibria in competitive societies, with applications to facility location, traffic routing and auctions,” in *The 43rd Annual IEEE Symposium on Foundations of Computer Science, 2002. Proceedings.*, Nov 2002, pp. 416–425.

[3] T. Murray, J. Garg, and R. Nagi, “Limited-trust equilibria,” *European Journal of Operational Research*, vol. 289, no. 1, pp. 364 – 380, 2021. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0377221720306202

[4] W. D. Hamilton, “The evolution of altruistic behavior,” *The American Naturalist*, vol. 97, no. 896, pp. 354–356, 1963.

[5] W. D. Hamilton, “The genetical evolution of social behaviour. i,” *Journal of Theoretical Biology*, vol. 7, no. 1, pp. 1 – 16, 1964.

[6] W. D. Hamilton, “The genetical evolution of social behaviour. ii,” *Journal of Theoretical Biology*, vol. 7, pp. 17 – 52, 1964.

[7] J. Q. Wilson and G. L. Kelling, “Broken windows,” *Atlantic monthly*, vol. 249, no. 3, pp. 29–38, 1982.

[8] D. Fudenberg and E. Maskin, “The folk theorem in repeated games with discounting or with incomplete information,” *Econometrica*, vol. 54, no. 3, pp. 533–554, 1986.

[9] J.-P. Benoit and V. Krishna, “Finitely repeated games,” *Econometrica*, vol. 53, no. 4, pp. 905–922, 1985.

[10] R. J. Aumann and L. S. Shapley, *Long-Term Competition—A Game-Theoretic Analysis.* New York, NY: Springer New York, 1994, pp. 1–15.

[11] A. Rubinstein, “Equilibrium in supergames with the overtaking criterion,” *Journal of Economic Theory*, vol. 21, no. 1, pp. 1 – 9, 1979.

[12] A. Rubinstein, "Strong perfect equilibrium in supergames," *International Journal of Game Theory*, vol. 9, no. 1, pp. 1–12, Mar 1980.

[13] J. Engle-Warnick and R. L. Slonim, "Learning to trust in indefinitely repeated games," *Games and Economic Behavior*, vol. 54, no. 1, pp. 95 – 114, 2006.

[14] P. Dasgupta, "Trust as a commodity," *Trust: Making and breaking cooperative relations*, vol. 4, pp. 49–72, 2000.

[15] R. Gibbons, *Trust in Social Structures: Hobbes and Coase Meet Repeated Games*, ser. The Russell Sage Foundation Series on Trust, volume 2. New York: Russell Sage Foundation, 2001, pp. 332–353.

[16] U. Leopold-Wildburger, J. H. Schuetze, and A. Lafer, "Mccockerel measuring individual punishment and reciprocity in a simple value-laden dilemma game," *European Journal of Operational Research*, vol. 140, no. 2, pp. 241 – 248, 2002.

[17] V. Anderhub, D. Engelmann, and W. Güth, "An experimental study of the repeated trust game with incomplete information," *Journal of Economic Behavior & Organization*, vol. 48, no. 2, pp. 197 – 216, 2002.

[18] J. Berg, J. Dickhaut, and K. McCabe, "Trust, reciprocity, and social history," *Games and Economic Behavior*, vol. 10, no. 1, pp. 122 – 142, 1995.

[19] J. Engle-Warnick and R. L. Slonim, "The evolution of strategies in a repeated trust game," *Journal of Economic Behavior & Organization*, vol. 55, no. 4, pp. 553 – 573, 2004, trust and Trustworthiness.

[20] R. Bapna, L. Qui, and S. C. Rice, "Repeated interactions vs. social ties: Quantifying the economic value of trust, forgiveness, and reputation using a field experiment," *MIS Quarterly (Forthcoming)*, 9 2016.

[21] J. Engle-Warnick and R. L. Slonim, "Inferring repeated-game strategies from actions: evidence from trust game experiments," *Economic Theory*, vol. 28, no. 3, pp. 603–632, Aug 2006.

[22] A. Morreale, L. Mittone, and G. L. Nigro, "Risky choices in strategic environments: An experimental investigation of a real options game," *European Journal of Operational Research*, vol. 279, no. 1, pp. 143 – 158, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0377221719304175

[23] G. lo Nigro, A. Morreale, S. Robba, and P. Roma, "Biopharmaceutical alliance and competition: A real options games approach," *International Journal of Innovation Management*, vol. 17, no. 06, p. 1340023, 2013.

[24] S.-P. Chen and W.-Y. Wu, "A systematic procedure to evaluate an automobile manufacturer–distributor partnership," *European Journal of Operational Research*, vol. 205, no. 3, pp. 687 – 698, 2010.

[25] S. Du, T. Nie, C. Chu, and Y. Yu, "Reciprocal supply chain with intention," *European Journal of Operational Research*, vol. 239, no. 2, pp. 389 – 402, 2014.

[26] J. A. Niederhoff and P. Kouvelis, "Generous, spiteful, or profit maximizing suppliers in the wholesale price contract: A behavioral study," *European Journal of Operational Research*, vol. 253, no. 2, pp. 372 – 382, 2016.

[27] S. Choi and P. R. Messinger, "The role of fairness in competitive supply chain relationships: An experimental study," *European Journal of Operational Research*, vol. 251, no. 3, pp. 798 – 813, 2016.

[28] P. Dal Bó and G. R. Fréchette, "On the determinants of cooperation in infinitely repeated games: A survey," *Journal of Economic Literature*, vol. 56, no. 1, pp. 60–114, March 2018.

[29] J. O. Ledyard, "Public Goods: A Survey of Experimental Research," University Library of Munich, Germany, Public Economics 9405003, May 1994.

[30] C. Chen, S.-F. Cheng, and H. C. Lau, "Multi-agent orienteering problem with time-dependent capacity constraints," *Web Intelli. and Agent Sys.*, vol. 12, no. 4, pp. 347–358, Oct. 2014.

[31] I. Caragiannis, C. Kaklamanis, P. Kanellopoulos, M. Kyropoulou, and E. Papaioannou, "The impact of altruism on the efficiency of atomic congestion games," in *Trustworthy Global Computing*, M. Wirsing, M. Hofmann, and A. Rauschmayer, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 172–188.

[32] T. Roughgarden, "Intrinsic robustness of the price of anarchy," *J. ACM*, vol. 62, no. 5, pp. 32:1–32:42, Nov. 2015.

[33] S. Kakutani, "A generalization of brouwer's fixed point theorem," *Duke Math. J.*, vol. 8, no. 3, pp. 457–459, 09 1941.

[34] C. E. Lemke and J. T. Howson, "Equilibrium points of bimatrix games," *Journal of the Society for Industrial and Applied Mathematics*, vol. 12, no. 2, pp. 413–423, 1964.

[35] C. E. Lemke, "Bimatrix equilibrium points and mathematical programming," *Management Science*, vol. 11, no. 7, pp. 681–689, 1965.

[36] S. Govindan and R. Wilson, "A global newton method to compute nash equilibria," *Journal of Economic Theory*, vol. 110, no. 1, pp. 65 – 86, 2003. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S002205310300005X

[37] X. Chen and X. Deng, "Settling the complexity of two-player nash equilibrium," in *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science*, ser. FOCS '06. USA: IEEE Computer Society, 2006, p. 261–272.

[38] P. Barclay, "Trustworthiness and competitive altruism can also solve the "tragedy of the commons"," *Evolution and Human Behavior*, vol. 25, no. 4, pp. 209 – 220, 2004. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S109051380400025X

[39] P. Barclay and R. Willer, "Partner choice creates competitive altruism in humans," *Proceedings. Biological sciences / The Royal Society*, vol. 274, pp. 749–53, 04 2007.

[40] K. Sylwester and G. Roberts, "Cooperators benefit through reputation-based partner choice in economic games," *Biology letters*, vol. 6, pp. 659–62, 10 2010.

[41] K. Sylwester and G. Roberts, "Reputation-based partner choice is an effective alternative to indirect reciprocity in solving social dilemmas," *Evolution and Human Behavior*, vol. 34, no. 3, pp. 201 – 206, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1090513812001250

[42] P. Barclay, "Strategies for cooperation in biological markets, especially for humans," *Evolution and Human Behavior*, vol. 34, no. 3, pp. 164 – 175, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1090513813000214

[43] P. Barclay, "Biological markets and the effects of partner choice on cooperation and friendship," *Current Opinion in Psychology*, vol. 7, pp. 33 – 38, 2016, evolutionary psychology. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2352250X15001906

[44] S. Debove, J.-B. André, and N. Baumard, "Partner choice creates fairness in humans," *Proceedings. Biological sciences / The Royal Society*, vol. 282, 06 2015.

[45] A. Eisenbruch, R. Grillot, and J. Roney, "Why be generous? tests of the partner choice and threat premium models of resource division," *Adaptive Human Behavior and Physiology*, vol. 5, 09 2019.

[46] G. Abramson and M. Kuperman, "Social games in a social network," *Phys. Rev. E*, vol. 63, p. 030901, Feb 2001. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevE.63.030901

[47] N. Hanaki, A. Peterhansl, P. Dodds, and D. Watts, "Cooperation in evolving social networks," *Management Science*, vol. 53, pp. 1036–1050, 07 2007.

[48] L. Dall'Asta, M. Marsili, and P. Pin, "Collaboration in social networks," *Proceedings of the National Academy of Sciences*, vol. 109, no. 12, pp. 4395–4400, 2012. [Online]. Available: https://www.pnas.org/content/109/12/4395

[49] E. Ozkan-Canbolat and A. Beraha, "Evolutionary knowledge games in social networks," *Journal of Business Research*, vol. 69, no. 5, pp. 1807 – 1811, 2016, designing implementable innovative realities. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S014829631500483X

[50] S. Bolouki, A. Nedić, and T. Başar, *Social Networks*. Cham: Springer International Publishing, 2018, pp. 907–949. [Online]. Available: https://doi.org/10.1007/978-3-319-44374-4_32

[51] L. Jain, R. Katarya, and S. Sachdeva, "Recognition of opinion leaders coalitions in online social network using game theory," *Knowledge-Based Systems*, vol. 203, p. 106158, 2020. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0950705120304007

[52] M. Scatà, A. Di Stefano, A. La Corte, P. Liò, E. Catania, E. Guardo, and S. Pagano, "Combining evolutionary game theory and network theory to analyze human cooperation patterns," *Chaos, Solitons & Fractals*, vol. 91, pp. 17 – 24, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0960077916301497

[53] S. Aral and D. Walker, "Tie strength, embeddedness, and social influence: A large-scale networked experiment," *Management Science*, vol. 60, pp. 1352–1370, 06 2014.

[54] G. Szabó and G. Fáth, "Evolutionary games on graphs," *Physics Reports*, vol. 446, no. 4, pp. 97 – 216, 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0370157307001810

[55] M. O. Jackson and Y. Zenou, "Chapter 3 - games on networks," in *Handbook of Game Theory with Economic Applications*, H. P. Young and S. Zamir, Eds. Elsevier, 2015, vol. 4, pp. 95 – 163. [Online]. Available: http://www.sciencedirect.com/science/article/pii/B9780444537669000033

[56] F. Fu, C. Hauert, M. A. Nowak, and L. Wang, "Reputation-based partner choice promotes cooperation in social networks," *Phys. Rev. E*, vol. 78, p. 026117, Aug 2008. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevE.78.026117

[57] P. Dasgupta and E. Maskin, "The existence of equilibrium in discontinuous economic games, ii: Applications," *The Review of Economic Studies*, vol. 53, no. 1, pp. 27–41, 1986. [Online]. Available: http://www.jstor.org/stable/2297589

[58] W. W. Zachary, "An information flow model for conflict and fission in small groups," *Journal of Anthropological Research*, vol. 33, no. 4, pp. 452–473, 1977.

[59] T. Tsiligirides, "Heuristic methods applied to orienteering," *The Journal of the Operational Research Society*, vol. 35, no. 9, pp. 797–809, 1984.

[60] C. Archetti, A. Hertz, and M. G. Speranza, "Metaheuristics for the team orienteering problem," *Journal of Heuristics*, vol. 13, no. 1, pp. 49–76, Feb 2007.

[61] S. Boussier, D. Feillet, and M. Gendreau, "An exact algorithm for team orienteering problems," *4OR*, vol. 5, no. 3, pp. 211–230, Sep 2007.

[62] I.-M. Chao, B. L. Golden, and E. A. Wasil, "The team orienteering problem," *European Journal of Operational Research*, vol. 88, no. 3, pp. 464 – 474, 1996.

[63] R. El-Hajj, D. Dang, and A. Moukrim, "Solving the team orienteering problem with cutting planes," *CoRR*, vol. abs/1604.02934, 2016.

[64] H. Tang and E. Miller-Hooks, "A tabu search heuristic for the team orienteering problem," *Computers & Operations Research*, vol. 32, no. 6, pp. 1379 – 1407, 2005.

[65] P. Vansteenwegen, W. Souffriau, G. V. Berghe, and D. V. Oudheusden, "A guided local search metaheuristic for the team orienteering problem," *European Journal of Operational Research*, vol. 196, no. 1, pp. 118 – 127, 2009.

[66] G. Best and G. A. Hollinger, "Decentralised self-organising maps for the online orienteering problem with neighbourhoods," in *International Symposium on Multi-Robot and Multi-Agent Systems*, 2019, pp. 139–141.

[67] G. Laporte and S. Martello, "The selective travelling salesman problem," *Discrete Applied Mathematics*, vol. 26, no. 2, pp. 193 – 207, 1990.

[68] V. K. Shetty, M. Sudit, and R. Nagi, "Priority-based assignment and routing of a fleet of unmanned combat aerial vehicles," *Computers & Operations Research*, vol. 35, no. 6, pp. 1813 – 1828, 2008, part Special Issue: OR Applications in the Military and in Counter-Terrorism.

[69] F. Mufalli, R. Batta, and R. Nagi, "Simultaneous sensor selection and routing of unmanned aerial vehicles for complex mission plans," *Computers & Operations Research*, vol. 39, no. 11, pp. 2787 – 2799, 2012.

[70] Y. Xia, R. Batta, and R. Nagi, "Controlling a fleet of unmanned aerial vehicles to collect uncertain information in a threat environment," *Oper. Res.*, vol. 65, no. 3, pp. 674–692, June 2017.

[71] C. Phan and H. Liu, "A cooperative uav/ugv platform for wildfire detection and fighting," in *System Simulation and Scientific Computing, 2008. ICSC 2008. Asia Simulation Conference-7th International Conference on.* IEEE, 2008, pp. 494–498.

[72] J. Tisdale, A. Ryan, Z. Kim, D. Tornqvist, and J. Hedrick, "A multiple uav system for vision-based search and localization," in *American Control Conference, 2008.* IEEE, 2008, pp. 1985–1990.

[73] M. Alighanbari and J. How, "Decentralized task assignment for unmanned aerial vehicles," in *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05. 44th IEEE Conference on.* IEEE, 2005, pp. 5668–5673.

[74] Y. Yang, A. Minai, and M. Polycarpou, "Decentralized cooperative search by networked uavs in an uncertain environment," in *American Control Conference*, vol. 6. IEEE, 2004, pp. 5558–5563.

[75] J. Tisdale, Z. Kim, and J. Hedrick, "Autonomous uav path planning and estimation," *Robotics & Automation Magazine, IEEE*, vol. 16, no. 2, pp. 35–42, 2009.

[76] M. Han, Y.-B. Wang, X. Chen, and X. Zhao, "A research on vehicle and UAV routing problem during distribution based on IAMMAS," *IOP Conference Series: Materials Science and Engineering*, vol. 688, p. 044004, 2019.

[77] A. Thibbotuwawa, G. Bocewicz, P. Nielsen, and B. Zbigniew, "Planning deliveries with UAV routing under weather forecast and energy consumption constraints," *IFAC-PapersOnLine*, vol. 52, no. 13, pp. 820 – 825, 2019, 9th IFAC Conference on Manufacturing Modelling, Management and Control MIM 2019.

[78] J. Baek, S. I. Han, and Y. Han, "Energy-efficient UAV routing for wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 2, pp. 1741–1750, 2020.

[79] O. Thakoor, J. Garg, and R. Nagi, "Multiagent uav routing: A game theory analysis with tight price of anarchy bounds," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 1, pp. 100–116, January 2020.

[80] R. W. Rosenthal, "A class of games possessing pure-strategy nash equilibria," *International Journal of Game Theory*, vol. 2, no. 1, pp. 65–67, Dec 1973.

[81] T. Roughgarden, "Selfish routing," Ph.D. dissertation, Cornell University, 2002.

[82] J. Correa and N. E. Stier-Moses, "Stackelberg routing in atomic network games," *Technical Report, Columbia Business School*, February 2007.

[83] G. Karakostas and S. G. Kolliopoulos, "Stackelberg strategies for selfish routing ingeneralmulticommodity networks," *Algorithmica*, vol. 53, no. 1, pp. 132–153, Jan 2009.

[84] L. Fang, K. Hipel, and D. Kilgour, *Interactive Decision Making: The Graph Model for Conflict Resolution*, ser. A Wiley-Interscience publication. Wiley, 1993, no. v. 3.

[85] P. Vansteenwegen, W. Souffriau, and D. V. Oudheusden, "The orienteering problem: A survey," *European Journal of Operational Research*, vol. 209, no. 1, pp. 1 – 10, 2011.

[86] J. Yen, "An algorithm for finding shortest routes from all source nodes to a given destination in general networks," *Quarterly of Applied Mathematics*, vol. 27, 01 1970.

[87] D. R. Shier, "On algorithms for finding the k shortest paths in a network," *Networks*, vol. 9, no. 3, pp. 195–214, 1979.